Maple computations

Contents

1 The example of $\mathbb{P}^2$ blown up in 15 points

```
> restart:
> with(LinearAlgebra):
```

1.1 First definitions

```
> restart:
> with(LinearAlgebra):
```

The function `f` is the function ϕ of the text given in the affine coordinates (y, z).

```
> f1:=(y,z)->(y*z^2)/(z^2+y^3):
> f2:=(y,z)->(z^3)/(z^2+y^3):
```

```
> f:=(y,z)->(f1(y,z),f2(y,z)):
```

The inverse of `f` is denoted by `g`.

```
> g1:=(y,z)->y*z^2/(z^2-y^3):
> g2:=(y,z)->z^3/(z^2-y^3):
> g:=(y,z)->(g1(y,z),g2(y,z)):
```

We define the action of the matrix `A` in the coordinates (y, z), its inverse is `B`.

```
> A:=(y,z)->((-1+(1+alpha)*z)/(alpha+2*(1-alpha)*y+(2+alpha-alpha^2)*z),(1-2*y+(1-alpha)*
z)/(alpha+2*(1-alpha)*y+(2+alpha-alpha^2)*z)):
> B:=(y,z)->((1-y-(1+alpha)*z)/(1+alpha+(alpha-3)*y+(1-alpha^2)*z),(1+y+(1-alpha)*z)/(1+alpha+
(alpha-3)*y+(1-alpha^2)*z)):
```

We define `m` = `AfAfA` and `n` = `BgBgB`.

```
> m1:=(y,z)->op(1,[A(f(A(f(A(y,z)))))]):
> m2:=(y,z)->op(2,[A(f(A(f(A(y,z)))))]):
> m:=(y,z)->(m1(y,z), m2(y,z)):
> n1:=(y,z)->op(1,[B(g(B(g(B(y,z)))))]):
> n2:=(y,z)->op(2,[B(g(B(g(B(y,z)))))]):
> n:=(y,z)->(n1(y,z), n2(y,z)):
```

1.2 The procedures to compute direct images of vector fields

1.2.1 The procedures `champ`

The procedure `champ1` associates to a vector field Z the coordinates of the Taylor expansion of the vector field $A_* Z$ in the following order: ∂_y, $y\,\partial_y$, $z\,\partial_y$, ∂_z, $y\,\partial_z$, $z\,\partial_z$, $y^2\,\partial_z$. The variable of the procedure is not the vector field itself but a curve $t \mapsto M_t$ whose the derivate at 0 is Z. For example, if $Z = u(y,z)\partial_y + v(y,z)\partial_z$, we can take $M_t(y,z) = (u(y,z) + ty, v(y,z) + tz)$.

```
> champ1:=proc(M)
> local prod,res,res1,res2,u,v:
> global resu1,resu2,w:
> prod:=unapply([(A@(M@B))(y,z)],t):
> res:=simplify(subs(t=0,diff(prod(t),t))):
> res1:=unapply(op(1,res),(y,z)):
> res2:=unapply(op(2,res),(y,z)):
> resu1:=[subs({y=0,z=0},res1(y,z)),subs({y=0,z=0},diff(res1(y,z),y)),subs({y=0,z=0},diff(
res1(y,z),z))]:
> resu2:=[subs({y=0,z=0},res2(y,z)),subs({y=0,z=0},diff(res2(y,z),y)),subs({y=0,z=0},diff(
res2(y,z),z)),subs({y=0,z=0},diff(diff(res2(y,z),y),y)/2)]:
> u:=op(resu1):v:=op(resu2):
> w:=[u,v]:
> RETURN(convert(w,Vector))
> end:
```

The procedure `champ2` associates to a vector field Z the coordinates of the Taylor expansion of the vector field $(\phi A)_*^{-1} Z$ in the following order: ∂_y, $y\,\partial_y$, $z\,\partial_y$, ∂_z, $y\,\partial_z$, $z\,\partial_z$, $y^2\,\partial_z$.

```
> champ2:=proc(M)
> local prod,res,res1,res2,u,v:
> global resu1,resu2,w:
> prod:=unapply([(B@(g@(M@(f@A))))(y,z)],t):
> res:=simplify(subs(t=0,diff(prod(t),t))):
> res1:=unapply(op(1,res),(y,z)):
> res2:=unapply(op(2,res),(y,z)):
> resu1:=[subs({y=0,z=0},res1(y,z)),subs({y=0,z=0},diff(res1(y,z),y)),subs({y=0,z=0},diff(
res1(y,z),z))]:
> resu2:=[subs({y=0,z=0},res2(y,z)),subs({y=0,z=0},diff(res2(y,z),y)),subs({y=0,z=0},diff(
res2(y,z),z)),subs({y=0,z=0},diff(diff(res2(y,z),y),y)/2)]:
> u:=op(resu1):v:=op(resu2):
> w:=[u,v]:
> RETURN(convert(w,Vector))
> end:
```

The procedure `champ3` associates to a vector field Z the coordinates of the Taylor expansion of the vector field $(A\phi A)_*^{-1}(Z)$ in the following order: ∂_y, $y\,\partial_y$, $z\,\partial_y$, ∂_z, $y\,\partial_z$, $z\,\partial_z$, $y^2\,\partial_z$.

```
> champ3:=proc(M)
```

```
>  local prod,res,res1,res2:
>  global resu1,resu2:
>  prod:=unapply([(B@(g@(B@(M@(A@(f@A))))))(y,z)],t):
>  res:=simplify(subs(t=0,diff(prod(t),t))):
>  res1:=unapply(op(1,res),(y,z)):
>  res2:=unapply(op(2,res),(y,z)):
>  resu1:=[subs({y=0,z=0},res1(y,z)),subs({y=0,z=0},diff(res1(y,z),y)),subs({y=0,z=0},diff(
res1(y,z),z))]:
>  resu2:=[subs({y=0,z=0},res2(y,z)),subs({y=0,z=0},diff(res2(y,z),y)),subs({y=0,z=0},diff(
res2(y,z),z)),subs({y=0,z=0},diff(diff(res2(y,z),y),y)/2)]:
>  RETURN(Transpose(convert([seq(factor(op(k,resu1)),k=1..nops(resu1)),seq(factor(op(k,resu2)
),k=1..nops(resu2))],Matrix))):
>  end:
```

The procedure `champ4` associates to a vector field Z the coordinates of the Taylor expansion of the vector field $A_*^{-1}(Z)$ in the following order: ∂_y, $y\partial_y$, $z\partial_y$, ∂_z, $y\partial_z$, $z\partial_z$, $y^2\partial_z$.

```
>  champ4:=proc(M)
>  local prod,res,res1,res2,u,v:
>  global resu1,resu2,w:
>  prod:=unapply([(B@(M@A))(y,z)],t):
>  res:=simplify(subs(t=0,diff(prod(t),t))):
>  res1:=unapply(op(1,res),(y,z)):
>  res2:=unapply(op(2,res),(y,z)):
>  resu1:=[subs({y=0,z=0},res1(y,z)),subs({y=0,z=0},diff(res1(y,z),y)),subs({y=0,z=0},diff(
res1(y,z),z))]:
>  resu2:=[subs({y=0,z=0},res2(y,z)),subs({y=0,z=0},diff(res2(y,z),y)),subs({y=0,z=0},diff(
res2(y,z),z)),subs({y=0,z=0},diff(diff(res2(y,z),y),y)/2)]:
>  u:=op(resu1):v:=op(resu2):
>  w:=[u,v]:
>  RETURN(convert(w,Vector))
>  end:
```

The procedure `champ5` associates to a vector field Z the coordinates of the Taylor expansion of the vector field $(A\phi A\phi A)_*^{-1}(Z)$ in the following order: ∂_y, $y\,\partial_y$, $z\,\partial_y$, ∂_z, $y\,\partial_z$, $z\,\partial_z$, $y^2\,\partial_z$.

```
>  champ5:=proc(M)
>  local prod,res,res1,res2,u,v:
>  global resu1,resu2,w:
>  prod:=unapply([(m@(M@n))(y,z)],t):
>  res:=(subs(t=0,diff(prod(t),t))):
>  res1:=unapply(op(1,res),(y,z)):
>  res2:=unapply(op(2,res),(y,z)):
>  resu1:=[subs({y=0,z=0},res1(y,z)),subs({y=0,z=0},diff(res1(y,z),y)),subs({y=0,z=0},diff(
res1(y,z),z))]:
>  resu2:=[subs({y=0,z=0},res2(y,z)),subs({y=0,z=0},diff(res2(y,z),y)),subs({y=0,z=0},diff(
res2(y,z),z)),subs({y=0,z=0},diff(diff(res2(y,z),y),y)/2)]:
>  u:=op(resu1):v:=op(resu2):
>  w:=[u,v]:
>  RETURN(convert(w,Vector))
>  end:
```

This procedure is too heavy for Maple, we will not use it. To replace it, we will do successive Taylor expansions at each step of blow-ups. We deal with it in the next section.

1.2.2 The procedures `devlim` and `extract`

We compute the points where we take the Taylor expansions.

```
>  A(0,0);
```

$-\alpha^{-1}$, α^{-1}

```
>  simplify(op(1,[(A@f@A)(0,0)])), simplify(op(2,[(A@f@A)(0,0)]));
```

α^{-1}, α^{-1}

```
>  simplify(op(1,[(A@f@A@f@A)(0,0)])), simplify(op(2,[(A@f@A@f@A)(0,0)]));
```

0, 0

The procedure `devlim1` associates to a vector field Z in coordinates (y,z) (still given by the derivative at 0 of a curve M_t) the Taylor expansion of A_*Z of order 2 at a point (p,q). The result is viewed as a polynomial vector field in a neighborhood of (p,q) and the procedure returns a curve N_t associated to this vector field.

```
> devlim1:=proc(M,p,q)
> local prod,res,res1,res2,resu1,resu2,N,MMM:
> MMM:=(y,z)->(op(M(y,z))):
> prod:=unapply([(A@(MMM@B))(y,z)],t):
> res:=simplify(subs(t=0,diff(prod(t),t))):
> res1:=unapply(op(1,res),(y,z)):
> res2:=unapply(op(2,res),(y,z)):
> resu1:=simplify(mtaylor(res1(y,z),[y=p,z=q],3)):
> resu2:=simplify(mtaylor(res2(y,z),[y=p,z=q],3)):
> N:=unapply([y+t*resu1,z+t*resu2],(y,z)):
> RETURN(N):
> end:
```

The procedure `devlim2` is similar to devlim1 but computes the Taylor expansion of $(A\phi)_*Z$ instead of the Taylor expansion of A_*Z.

```
> devlim2:=proc(M,p,q)
> local prod,res,res1,res2,resu1,resu2,N,MMM:
> MMM:=(y,z)->(op(M(y,z))):
> prod:=unapply([(A@f@(MMM@g@B))(y,z)],t):
> res:=simplify(subs(t=0,diff(prod(t),t))):
> res1:=unapply(op(1,res),(y,z)):
> res2:=unapply(op(2,res),(y,z)):
> resu1:=simplify(mtaylor(res1(y,z),[y=p,z=q],3)):
> resu2:=simplify(mtaylor(res2(y,z),[y=p,z=q],3)):
> N:=unapply([y+t*resu1,z+t*resu2],(y,z)):
> RETURN(N):
> end:
```

The procedure `extract` associates to a vector field given by a curve $N_t(y,z)$ the components in ∂_y, $y\partial_y$, $z\partial_y$, ∂_z, $y\partial_z$, $z\partial_z$ and $y^2\partial_z$ in the Taylor expansion of the associated vector field at $(0,0)$.

```
> extract:=proc(M)
> local prod,res,res1,res2,w:
> prod:=unapply(M(y,z),t):
> res:=simplify(subs(t=0,diff(prod(t),t))):
> res1:=unapply(op(1,res),(y,z)):
> res2:=unapply(op(2,res),(y,z)):
> w[1]:=simplify(res1(0,0)):
> w[2]:=simplify(subs({y=0,z=0},diff(res1(y,z),y))):
> w[3]:=simplify(subs({y=0,z=0},diff(res1(y,z),z))):
> w[4]:=simplify(res2(0,0)):
> w[5]:=simplify(subs({y=0,z=0},diff(res2(y,z),y))):
> w[6]:=simplify(subs({y=0,z=0},diff(res2(y,z),z))):
> w[7]:=simplify(subs({y=0,z=0},diff(res2(y,z),y$2)/2)):
> w:=[w[1],w[2],w[3],w[4],w[5],w[6],w[7]]:
> RETURN(convert(w,Vector)):
> end:
```

The function $\texttt{psi}_1$ is the projection of the blow-up of $(0,0)_{y,z}$, and $\texttt{eta}_1$ is its inverse.

```
> psi1:=(u,v)->(u,u*v):
> eta1:=(y,z)->(y,z/y):
```

The function $\texttt{psi}_2$ is the projection of the blow-up of $(-1/\alpha, 1/\alpha)_{y,z}$ and $\texttt{eta}_2$ is its inverse.

```
> psi2:=(u,v)->(u-1/alpha,u*v+1/alpha):
> eta2:=(y,z)->(y+1/alpha,(z-1/alpha)/(y+1/alpha)):
```

The function $\texttt{psi}_3$ is the projection of the blow-up of $(1/\alpha, 1/\alpha)_{y,z}$ and $\texttt{eta}_3$ is its inverse.

```
> psi3:=(u,v)->(u+1/alpha, u*v+1/alpha):
> eta3:=(y,z)->(y-1/alpha,(z-1/alpha)/(y-1/alpha)):
```

The procedure `u1devlim1` associates to a vector field Z on $\mathbb{P}^2$ blown up at $(0,0)_{y,z}$ written in the coordinates (u_1,v_1) the Taylor expansion of A_*Z of order 3 at a point (p,q). The result is considered as a polynomial vector field in a neighborhood of (p,q) in the coordinates (u_1,v_1) which are this time the coordinates on $\mathbb{P}^2$ blown up at $\left(-\frac{1}{\alpha},\frac{1}{\alpha}\right)_{y,z}$, and the procedure gives a curve N_t associated to this vector field.

```
> u1devlim1:=proc(M,p,q)
> local prod,res,res1,res2,resu1,resu2,N,MMM:
> MMM:=(u,v)->(op(M(u,v))):
> prod:=unapply([(eta2@(A@(psi1@MMM@eta1@(B@psi2))))(u,v)],t):
> res:=simplify(subs(t=0,diff(prod(t),t))):
```

```
>  res1:=unapply(simplify(op(1,res)),(u,v)):
>  res2:=unapply(simplify(op(2,res)),(u,v)):
>  resu1:=simplify(mtaylor(res1(u,v),[u=p,v=q],4)):
>  resu2:=simplify(mtaylor(res2(u,v),[u=p,v=q],4)):
>  N:=unapply([u+t*resu1,v+t*resu2],(u,v)):
>  RETURN(N):
>  end:
```

We compute the points where we take the Taylor expansions.

```
>  limit(op(1,[(eta2@(A@psi1))(u,v)]),{u=0,v=0}), limit(op(2,[(eta2@(A@psi1))(u,v)]),{u=0,
v=0});
```

$0,\ -(1-\alpha)^{-1}$

```
>  limit(op(1,[(eta3@(A@f@A)@psi1)(u,v)]),{u=0,v=0}), limit(op(2,[(eta3@(A@f@A)@psi1)(u,v)
]),{u=0,v=0});
```

$0,\ -(-1-\alpha)^{-1}$

```
>  limit(op(1,[(eta1@(A@f@A@f@A)@psi1)(u,v)]),{u=0,v=0}), limit(op(2,[(eta1@(A@f@A@f@A)@psi1)
(u,v)]),{u=0,v=0});
```

$0,\ 0$

The procedure `u1devlim2` associates to a vector field Z on the blow up of $\mathbb{P}^2$ at $\left(-\frac{1}{\alpha},\frac{1}{\alpha}\right)_{y,z}$ expressed in the coordinates (u_1,v_1) the Taylor expansion of $(A\phi)_*\,Z$ of order 3 at a point (p,q). The result is viewed as a polynomial vector field in a neighborhood of (p,q) in the coordinates (u_1,v_1) which are this time some coordinates on $\mathbb{P}^2$ blown up at $\left(\frac{1}{\alpha},\frac{1}{\alpha}\right)_{y,z}$, and the procedure returns a curve N_t associated to this vector field.

```
>  u1devlim2:=proc(M,p,q)
>  local prod,res,res1,res2,resu1,resu2,N,MMM:
>  MMM:=(u,v)->(op(M(u,v))):
>  prod:=unapply([(eta3@(A@f@(psi2@MMM@eta2@g@B@psi3)))(u,v)],t):
>  res:=simplify(subs(t=0,diff(prod(t),t))):
>  res1:=unapply(simplify(op(1,res)),(u,v)):
>  res2:=unapply(simplify(op(2,res)),(u,v)):
>  resu1:=simplify(mtaylor(res1(u,v),[u=p,v=q],4)):
>  resu2:=simplify(mtaylor(res2(u,v),[u=p,v=q],4)):
>  N:=unapply([u+t*resu1,v+t*resu2],(u,v)):
>  RETURN(N):
>  end:
```

The procedure `u1devlim3` associates to a vector field Z on the blow up of $\mathbb{P}^2$ at $\left(\frac{1}{\alpha},\frac{1}{\alpha}\right)_{y,z}$ expressed in the coordinates (u_1,v_1) the Taylor expansion of $(A\phi)_*\,Z$ of order 3 at a point (p,q). The result is viewed as a polynomial vector field in a neighborhood of (p,q) in the coordinates (u_1,v_1) which are this time coordinates on $\mathbb{P}^2$ blown up at $(0,0)_{y,z}$, and the procedure gives a curve N_t associated to this vector field.

```
>  u1devlim3:=proc(M,p,q)
>  local prod,res,res1,res2,resu1,resu2,N,MMM:
>  MMM:=(u,v)->(op(M(u,v))):
>  prod:=unapply([(eta1@(A@f@(psi3@MMM@eta3@g@B@psi1)))(u,v)],t):
>  res:=simplify(subs(t=0,diff(prod(t),t))):
>  res1:=unapply(simplify(op(1,res)),(u,v)):
>  res2:=unapply(simplify(op(2,res)),(u,v)):
>  resu1:=simplify(mtaylor(res1(u,v),[u=p,v=q],4)):
>  resu2:=simplify(mtaylor(res2(u,v),[u=p,v=q],4)):
>  N:=unapply([u+t*resu1,v+t*resu2],(u,v)):
>  RETURN(N):
>  end:
```

The procedure `u1extract` associates to a vector field given by a curve $N_t(u_1,v_1)$ the components in ∂_{u_1}, $u_1\,\partial_{u_1}$, $v_1\,\partial_{u_1}$, $u_1v_1\,\partial_{u_1}$, $v_1^2\,\partial_{u_1}$, $v_1^3\,\partial_{u_1}$, ∂_{v_1}, $u_1\,\partial_{v_1}$, $v_1\,\partial_{v_1}$ and $v_1^2\,\partial_{v_1}$ of the Taylor expansion of the associated vector field at $(0,0)$.

```
>  u1extract:=proc(M)
>  local prod,res,res1,res2,w:
>  prod:=unapply(M(u,v),t):
>  res:=simplify(subs(t=0,diff(prod(t),t))):
>  res1:=unapply(op(1,res),(u,v)):
>  res2:=unapply(op(2,res),(u,v)):
>  w[1]:=simplify(res1(0,0)):
>  w[2]:=simplify(subs({u=0,v=0},diff(res1(u,v),u))):
>  w[3]:=simplify(subs({u=0,v=0},diff(res1(u,v),v))):
```

```
>  w[4]:=simplify(subs({u=0,v=0},diff(res1(u,v),u,v))):
>  w[5]:=simplify(subs({u=0,v=0},diff(res1(u,v),v$2))/2):
>  w[6]:=simplify(subs({u=0,v=0},diff(res1(u,v),v$3))/3):
>  w[7]:=simplify(res2(0,0)):
>  w[8]:=simplify(subs({u=0,v=0},diff(res2(u,v),u))):
>  w[9]:=simplify(subs({u=0,v=0},diff(res2(u,v),v))):
>  w[10]:=simplify(subs({u=0,v=0},diff(res2(u,v),v$2)/2)):
>  w:=[w[1],w[2],w[3],w[4],w[5],w[6],w[7],w[8],w[9],w[10]]:
>  RETURN(convert(w,Vector)):
>  end:
```

The function `ppsi`$_1$ is the projection of the blow up of $(0,0)_{y,z}$ and $(0,0)_{u_1,v_1}$, and `eeta`$_1$ is its inverse.

```
>  ppsi1:=(r,s)->(r*s,r*s^2):
>  eeta1:=(y,z)->(y^2/z,z/y):
```

The function `ppsi`$_2$ is the projection of the blow up of $\left(-\frac{1}{\alpha},\frac{1}{\alpha}\right)_{y,z}$ and $\left(0,\frac{1}{\alpha-1}\right)_{u_1,v_1}$, and `eeta`$_2$ is its inverse.

```
>  ppsi2:=(r,s)->(r*s-1/alpha, r*s*(s+1/(alpha-1))+1/alpha):
>  eeta2:=(y,z)->((y+1/alpha)/((z-1/alpha)/(y+1/alpha)-1/(alpha-1)),(z-1/alpha)/(y+1/alpha)
-1/(alpha-1)):
```

The function `ppsi`$_3$ is the projection of the blow up of $\left(\frac{1}{\alpha},\frac{1}{\alpha}\right)_{y,z}$ and $\left(0,\frac{1}{\alpha+1}\right)_{u_1,v_1}$, and `eeta`$_3$ is its inverse.

```
>  ppsi3:=(r,s)->(r*s+1/alpha, r*s*(s+1/(alpha+1))+1/alpha):
>  eeta3:=(y,z)->((y-1/alpha)/((z-1/alpha)/(y-1/alpha)-1/(alpha+1)),(z-1/alpha)/(y-1/alpha)
-1/(alpha+1)):
```

We compute the points where we take the Taylor expansions.

```
>  limit(op(1,[(eeta2@(A@ppsi1))(r,s)]),{r=0,s=0}), limit(op(2,[(eeta2@(A@ppsi1))(r,s)]),{r=0,
s=0});
```

0, 0

```
>  limit(op(1,[(eeta3@(A@f@A)@ppsi1)(r,s)]),{r=0,s=0}), limit(op(2,[(eeta3@(A@f@A)@ppsi1)(
r,s)]),{r=0,s=0});
```

0, 0

```
>  limit(op(1,[(eeta1@(A@f@A@f@A)@ppsi1)(r,s)]),{r=0,s=0}), limit(op(2,[(eeta1@(A@f@A@f@A)
@ppsi1)(r,s)]),{r=0,s=0});
```

0, 0

The procedure `r2devlim1` associates to a vector field Z on the blow up of $\mathbb{P}^2$ at $(0,0)_{y,z}$ and $(0,0)_{u_1,v_1}$ expressed in coordinates (r_2,s_2) the Taylor expansion of A_*Z of order 2 at a point (p,q). The result is viewed as a polynomial vector field in a neighborhood of (p,q) in the coordinates (r_2,s_2) which are this time coordinates on $\mathbb{P}^2$ blown up at $\left(-\frac{1}{\alpha},\frac{1}{\alpha}\right)_{y,z}$ and $\left(0,\frac{1}{\alpha-1}\right)_{u_1,v_1}$, and the procedure returns a curve N_t associated to this vector field.

```
>  r2devlim1:=proc(M,p,q)
>  local prod,res,res1,res2,resu1,resu2,N,MMM:
>  MMM:=(r,s)->(op(M(r,s))):
>  prod:=unapply([(eeta2@(A@(ppsi1@MMM@eeta1@(B@ppsi2))))(r,s)],t):
>  res:=simplify(subs(t=0,diff(prod(t),t))):
>  res1:=unapply(simplify(op(1,res)),(r,s)):
>  res2:=unapply(simplify(op(2,res)),(r,s)):
>  resu1:=simplify(mtaylor(res1(r,s),[r=p,s=q],3)):
>  resu2:=simplify(mtaylor(res2(r,s),[r=p,s=q],3)):
>  N:=unapply([r+t*resu1,s+t*resu2],(r,s)):
>  RETURN(N):
>  end:
```

The procedure `r2devlim2` associates to a vector field Z on the blow up of $\mathbb{P}^2$ at $\left(-\frac{1}{\alpha},\frac{1}{\alpha}\right)_{y,z}$ and $\left(0,\frac{1}{\alpha-1}\right)_{u_1,v_1}$ expressed in coordinates (r_2,s_2) the Taylor expansion of $(A\phi)_*Z$ of order 2 at a point (p,q). The result is viewed as a polynomial vector field in a neighborhood of (p,q) in the coordinates (r_2,s_2) which are this time the coordinates of $\mathbb{P}^2$ blown up at $\left(\frac{1}{\alpha},\frac{1}{\alpha}\right)_{y,z}$ and $\left(0,\frac{1}{\alpha+1}\right)_{u_1,v_1}$, and the procedure returns a curve N_t associated to this vector field.

```
>  r2devlim2:=proc(M,p,q)
>  local prod,res,res1,res2,resu1,resu2,N,MMM:
```

```
> MMM:=(r,s)->(op(M(r,s))):
> prod:=unapply([(eeta3@(A@f@(ppsi2@MMM@eeta2@g@B@ppsi3)))(r,s)],t):
> res:=simplify(subs(t=0,diff(prod(t),t))):
> res1:=unapply(simplify(op(1,res)),(r,s)):
> res2:=unapply(simplify(op(2,res)),(r,s)):
> resu1:=simplify(mtaylor(res1(r,s),[r=p,s=q],3)):
> resu2:=simplify(mtaylor(res2(r,s),[r=p,s=q],3)):
> N:=unapply([r+t*resu1,s+t*resu2],(r,s)):
> RETURN(N):
> end:
```

The procedure `r2devlim3` associates to a vector field Z on the blow up of $\mathbb{P}^2$ at $\left(\frac{1}{\alpha},\frac{1}{\alpha}\right)_{y,z}$ and $\left(0,\frac{1}{\alpha+1}\right)_{u_1,v_1}$ expressed in coordinates (r_2,s_2) the Taylor expansion of $(A\phi)_*Z$ of order 2 at a point (p,q). The result is viewed as a polynomial vector field in a neighborhood of (p,q) in the coordinates (r_2,s_2) which are this time the coordinates of $\mathbb{P}^2$ blown up at $(0,0)_{y,z}$ and $(0,0)_{u_1,v_1}$, and the procedure returns a curve N_t associated to this vector field.

```
> r2devlim3:=proc(M,p,q)
> local prod,res,res1,res2,resu1,resu2,N,MMM:
> MMM:=(r,s)->(op(M(r,s))):
> prod:=unapply([(eeta1@(A@f@(ppsi3@MMM@eeta3@g@B@ppsi1)))(r,s)],t):
> res:=simplify(subs(t=0,diff(prod(t),t))):
> res1:=unapply(simplify(op(1,res)),(r,s)):
> res2:=unapply(simplify(op(2,res)),(r,s)):
> resu1:=simplify(mtaylor(res1(r,s),[r=p,s=q],3)):
> resu2:=simplify(mtaylor(res2(r,s),[r=p,s=q],3)):
> N:=unapply([r+t*resu1,s+t*resu2],(r,s)):
> RETURN(N):
> end:
```

The procedure `r2extract` associates to a vector field given by a curve $N_t(r_2,s_2)$ the components in ∂_{r_2}, $r_2\,\partial_{r_2}$, $s_2\,\partial_{r_2}$, $r_2s_2\,\partial_{r_2}$, $r_2^2\,\partial_{r_2}$, $s_2^2\,\partial_{r_2}$, ∂_{s_2}, $r_2\,\partial_{s_2}$, $s_2\,\partial_{s_2}$, $r_2s_2\,\partial_{s_2}$, $r_2^2\,\partial_{s_2}$ and $s_2^2\,\partial_{s_2}$ in the Taylor expansion of the associated vector field at $(0,0)$.

```
> r2extract:=proc(M)
> local prod,res,res1,res2,w:
> prod:=unapply(M(r,s),t):
> res:=simplify(subs(t=0,diff(prod(t),t))):
> res1:=unapply(op(1,res),(r,s)):
> res2:=unapply(op(2,res),(r,s)):
> w[1]:=simplify(res1(0,0)):
> w[2]:=simplify(subs({r=0,s=0},diff(res1(r,s),r))):
> w[3]:=simplify(subs({r=0,s=0},diff(res1(r,s),s))):
> w[4]:=simplify(subs({r=0,s=0},diff(res1(r,s),r,s))):
> w[5]:=simplify(subs({r=0,s=0},diff(res1(r,s),r$2))/2):
> w[6]:=simplify(subs({r=0,s=0},diff(res1(r,s),s$2))/2):
> w[7]:=simplify(res2(0,0)):
> w[8]:=simplify(subs({r=0,s=0},diff(res2(r,s),r))):
> w[9]:=simplify(subs({r=0,s=0},diff(res2(r,s),s))):
> w[10]:=simplify(subs({r=0,s=0},diff(res2(r,s),r,s))):
> w[11]:=simplify(subs({r=0,s=0},diff(res2(r,s),r$2))/2):
> w[12]:=simplify(subs({r=0,s=0},diff(res2(r,s),s$2))/2):
> w:=[w[1],w[2],w[3],w[4],w[5],w[6],w[7],w[8],w[9],w[10],w[11],w[12]]:
> RETURN(convert(w,Vector)):
> end:
```

The function $\texttt{pppsi}_1$ is the projection of the blow up of $(0,0)_{y,z}$, $(0,0)_{u_1,v_1}$ and $(0,0)_{r_2,s_2}$, and $\texttt{eeeta}_1$ is its inverse.

```
> pppsi1:=(r,s)->(r*s*s,r*s*s^2):
> eeeta1:=(y,z)->((y^2/z)/(z/y),(z/y)):
```

The function $\texttt{pppsi}_2$ is the projection of the blow up of $\left(-\frac{1}{\alpha},\frac{1}{\alpha}\right)_{y,z}$, $\left(0,\frac{1}{\alpha-1}\right)_{u_1,v_1}$ and $(0,0)_{r_2,s_2}$, and $\texttt{eeeta}_2$ is its inverse.

```
> pppsi2:=(r,s)->(r*s*s-1/alpha, r*s*s*(s+1/(alpha-1))+1/alpha):
> eeeta2:=(y,z)->(((y+1/alpha)/((z-1/alpha)/(y+1/alpha)-1/(alpha-1)))/((z-1/alpha)/(y+1/alpha)
-1/(alpha-1)),((z-1/alpha)/(y+1/alpha)-1/(alpha-1))):
```

The function $\texttt{pppsi}_3$ is the projection of the blow up of $\left(\frac{1}{\alpha},\frac{1}{\alpha}\right)_{y,z}$, $\left(0,\frac{1}{\alpha+1}\right)_{u_1,v_1}$ and $(0,0)_{r_2,s_2}$, and $\texttt{eeeta}_3$ is its inverse.

```
> pppsi3:=(r,s)->(r*s*s+1/alpha, r*s*s*(s+1/(alpha+1))+1/alpha):
```

```
> eeeta3:=(y,z)->(((y-1/alpha)/((z-1/alpha)/(y-1/alpha)-1/(alpha+1)))/((z-1/alpha)/(y-1/alpha)
-1/(alpha+1)),((z-1/alpha)/(y-1/alpha)-1/(alpha+1))):
```

We compute the points where we take the Taylor expansions.

```
> limit(op(1,[(eeeta2@(A@pppsi1))(r,s)]),{r=1,s=0}), limit(op(2,[(eeeta2@(A@pppsi1))(r,s)
]),{r=1,s=0});
```

$$\frac{-(\alpha-1)^5}{2\alpha^4},\ 0$$

```
> limit(op(1,[(eeeta3@(A@f@A)@pppsi1)(r,s)]),{r=1,s=0}), limit(op(2,[(eeeta3@(A@f@A)@pppsi1)
(r,s)]),{r=1,s=0});
```

$$\frac{(1+\alpha)^5}{2\alpha^4},\ 0$$

```
> limit(op(1,[(eeeta1@(A@f@A@f@A)@pppsi1)(r,s)]),{r=1,s=0}), limit(op(2,[(eeeta1@(A@f@A@f@A)
@pppsi1)(r,s)]),{r=1,s=0});
```

$$-1,\ 0$$

The procedure `r3devlim1` associates to a vector field Z on the blow up of $\mathbb{P}^2$ at $(0,0)_{y,z}$, $(0,0)_{u_1,v_1}$ and $(0,0)_{r_2,s_2}$ expressed in coordinates (r_3,s_3) the Taylor expansion of A_*Z of order 1 at a point (p,q). The result is viewed as a polynomial vector field in a neighborhood of (p,q) in the coordinates (r_3,s_3) which are this time the coordinates on $\mathbb{P}^2$ blown up at $\left(-\frac{1}{\alpha},\frac{1}{\alpha}\right)_{y,z}$, $\left(0,\frac{1}{\alpha-1}\right)_{u_1,v_1}$ and $(0,0)_{r_2,s_2}$, and the procedure gives a curve N_t associated to this vector field.

```
> r3devlim1:=proc(M,p,q)
> local prod,res,res1,res2,resu1,resu2,N,MMM:
> MMM:=(r,s)->(op(M(r,s))):
> prod:=unapply([(eeeta2@(A@(pppsi1@MMM@eeeta1@(B@pppsi2))))(r,s)],t):
> res:=simplify(subs(t=0,diff(prod(t),t))):
> res1:=unapply(simplify(op(1,res)),(r,s)):
> res2:=unapply(simplify(op(2,res)),(r,s)):
> resu1:=simplify(mtaylor(res1(r,s),[r=p,s=q],2)):
> resu2:=simplify(mtaylor(res2(r,s),[r=p,s=q],2)):
> N:=unapply([r+t*resu1,s+t*resu2],(r,s)):
> RETURN(N):
> end:
```

The procedure `r3devlim2` associates to a vector field Z on the blow up of $\mathbb{P}^2$ at $\left(-\frac{1}{\alpha},\frac{1}{\alpha}\right)_{y,z}$, $\left(0,\frac{1}{\alpha-1}\right)_{u_1,v_1}$ and $(0,0)_{r_2,s_2}$ expressed in the coordinates (r_3,s_3) the Taylor expansion of $(A\phi)_*\,Z$ of order 1 at a point (p,q). The result is viewed as a polynomial vector field in a neighborhood of (p,q) in the coordinates (r_3,s_3) which are this time the coordinates of $\mathbb{P}^2$ blown up at $\left(\frac{1}{\alpha},\frac{1}{\alpha}\right)_{y,z}$, $\left(0,\frac{1}{\alpha+1}\right)_{u_1,v_1}$ and $(0,0)_{r_2,s_2}$, and the procedure returns a curve N_t associated to this vector field.

```
> r3devlim2:=proc(M,p,q)
> local prod,res,res1,res2,resu1,resu2,N,MMM:
> MMM:=(r,s)->(op(M(r,s))):
> prod:=unapply([(eeeta3@(A@f@(pppsi2@MMM@eeeta2@g@B@pppsi3)))(r,s)],t):
> res:=simplify(subs(t=0,diff(prod(t),t))):
> res1:=unapply(simplify(op(1,res)),(r,s)):
> res2:=unapply(simplify(op(2,res)),(r,s)):
> resu1:=simplify(mtaylor(res1(r,s),[r=p,s=q],2)):
> resu2:=simplify(mtaylor(res2(r,s),[r=p,s=q],2)):
> N:=unapply([r+t*resu1,s+t*resu2],(r,s)):
> RETURN(N):
> end:
```

The procedure `r3devlim3` associates to a vector field Z on the blow up of $\mathbb{P}^2$ at $\left(\frac{1}{\alpha},\frac{1}{\alpha}\right)_{y,z}$, $\left(0,\frac{1}{\alpha+1}\right)_{u_1,v_1}$ and $(0,0)_{r_2,s_2}$ expressed in coordinates (r_3,s_3) the Taylor expansion of $(A\phi)_*\,Z$ of order 1 at a point (p,q). The result is viewed as a polynomial vector field in a neighborhood of (p,q) in the coordinates (r_3,s_3) which are this time the coordinates on $\mathbb{P}^2$ blown up at $(0,0)_{y,z}$, $(0,0)_{u_1,v_1}$ and $(0,0)_{r_2,s_2}$, and the procedure returns a curve N_t associated to this vector field.

```
> r3devlim3:=proc(M,p,q)
> local prod,res,res1,res2,resu1,resu2,N,MMM:
> MMM:=(r,s)->(op(M(r,s))):
> prod:=unapply([(eeeta1@(A@f@(pppsi3@MMM@eeeta3@g@B@pppsi1)))(r,s)],t):
> res:=simplify(subs(t=0,diff(prod(t),t))):
> res1:=unapply(simplify(op(1,res)),(r,s)):
```

```
> res2:=unapply(simplify(op(2,res)),(r,s)):
> resu1:=simplify(mtaylor(res1(r,s),[r=p,s=q],2)):
> resu2:=simplify(mtaylor(res2(r,s),[r=p,s=q],2)):
> N:=unapply([r+t*resu1,s+t*resu2],(r,s)):
> RETURN(N):
> end:
```

The procedure `r3extract` associates to a vector field given by a curve $N_t(r_3, s_3)$ the components of ∂_{r_3}, $s_3\partial_{r_3}$ and ∂_{s_3} in the Taylor expansion of the associated vector field at $(0,0)$.

```
> r3extract:=proc(M)
> local prod,res,res1,res2,w:
> prod:=unapply(M(r,s),t):
> res:=simplify(subs(t=0,diff(prod(t),t))):
> res1:=unapply(op(1,res),(r,s)):
> res2:=unapply(op(2,res),(r,s)):
> w[1]:=simplify(res1(-1,0)):
> w[2]:=simplify(subs({r=-1,s=0},diff(res1(r,s),s))):
> w[3]:=simplify(res2(-1,0)):
> w:=[w[1],w[2],w[3]]:
> RETURN(convert(w,Vector)):
> end:
```

The function $\texttt{ppppsi}_1$ is the projection of the blow up of $(0,0)_{y,z}$, $(0,0)_{u_1,v_1}$ $(0,0)_{r_2,s_2}$ and $(1,0)_{r_3,s_3}$, and $\texttt{eeeeta}_1$ is its inverse.

```
> ppppsi1:=(r,s)->((r*s+1)*s*s,(r*s+1)*s*s^2):
> eeeeta1:=(y,z)->(((y^2/z)/(z/y)-1)/(z/y),(z/y)):
```

The function $\texttt{ppppsi}_2$ is the projection of the blow up at $\left(-\frac{1}{\alpha},\frac{1}{\alpha}\right)_{y,z}$, $\left(0,\frac{1}{\alpha-1}\right)_{u_1,v_1}$, $(0,0)_{r_2,s_2}$ and $\left(-\frac{(\alpha-1)^5}{2\alpha^4},0\right)_{r_3,s_3}$, and $\texttt{eeeeta}_2$ is its inverse.

```
> ppppsi2:=(r,s)->((r*s-(1/2)*(-1+alpha)^5/alpha^4)*s*s-1/alpha, (r*s-(1/2)*(-1+alpha)^5/
alpha^4)*s*s*(s+1/(alpha-1))+1/alpha):
> eeeeta2:=(y,z)->((((y+1/alpha)/((z-1/alpha)/(y+1/alpha)-1/(alpha-1)))/((z-1/alpha)/(y+1/
alpha)-1/(alpha-1))+(1/2)*(-1+alpha)^5/alpha^4)/((z-1/alpha)/(y+1/alpha)-1/(alpha-1)),((z-1/
alpha)/(y+1/alpha)-1/(alpha-1))):
```

The function $\texttt{ppppsi}_3$ is the projection of the blow up at $\left(\frac{1}{\alpha},\frac{1}{\alpha}\right)_{y,z}$, $\left(0,\frac{1}{\alpha+1}\right)_{u_1,v_1}$, $(0,0)_{r_2,s_2}$ and $\left(\frac{(\alpha+1)^5}{2\alpha^4},0\right)$, and $\texttt{eeeeta}_3$ is its inverse.

```
> ppppsi3:=(r,s)->((r*s+(1/2)*(1+alpha)^5/alpha^4)*s*s+1/alpha, (r*s+(1/2)*(1+alpha)^5/alpha^4)
*s*s*(s+1/(alpha+1))+1/alpha):
> eeeeta3:=(y,z)->((((y-1/alpha)/((z-1/alpha)/(y-1/alpha)-1/(alpha+1)))/((z-1/alpha)/(y-1/
alpha)-1/(alpha+1))-(1/2)*(1+alpha)^5/alpha^4)/((z-1/alpha)/(y-1/alpha)-1/(alpha+1)),((z-1/
alpha)/(y-1/alpha)-1/(alpha+1))):
```

The function $\texttt{ppppsi}_4$ is the projection of the blow up at $(0,0)_{y,z}$, $(0,0)_{u_1,v_1}$ $(0,0)_{r_2,s_2}$ and $(-1,0)_{r_3,s_3}$, and $\texttt{eeeeta}_1$ is its inverse.

```
> ppppsi4:=(r,s)->((r*s-1)*s*s,(r*s-1)*s*s^2):
> eeeeta4:=(y,z)->(((y^2/z)/(z/y)+1)/(z/y),(z/y)):
```

We compute the points where we take the Taylor expansions.

```
> limit(op(1,[(eeeeta2@(A@ppppsi1))(r,s)]),{r=0,s=0}), limit(op(2,[(eeeeta2@(A@ppppsi1))(
r,s)]),{r=0,s=0});
```

$$\frac{3(\alpha-1)^4(1-\alpha^2+\alpha^3-\alpha)}{4\alpha^5},\ 0$$

```
> limit(op(1,[(eeeeta3@(A@f@A)@ppppsi1)(r,s)]),{r=0,s=0}), limit(op(2,[(eeeeta3@(A@f@A)@ppppsi1)
(r,s)]),{r=0,s=0});
```

$$\frac{-3(1+\alpha)^4(-\alpha-1+\alpha^2+\alpha^3)}{4\alpha^5},\ 0$$

```
> limit(op(1,[(eeeeta4@(A@f@A@f@A)@ppppsi1)(r,s)]),{r=0,s=0}), limit(op(2,[(eeeeta4@(A@f@A@f@A)
@ppppsi1)(r,s)]),{r=0,s=0});
```

$$0,\ 0$$

The procedure `r4devlim1` associates to a vector field Z on the blow up of $\mathbb{P}^2$ at $(0,0)_{y,z}$, $(0,0)_{u_1,v_1}$, $(0,0)_{r_2,s_2}$ and $(1,0)_{r_3,s_3}$ expressed in the coordinates (r_4,s_4) the Taylor expansion of A_*Z of order 0 at a point (p,q). The result is viewed as a polynomial vector field in a neighborhood of (p,q) in the coordinates (r_4,s_4) which are

this time the coordinates on $\mathbb{P}^2$ blown up at $\left(-\frac{1}{\alpha},\frac{1}{\alpha}\right)_{y,z}$, $\left(0,\frac{1}{\alpha-1}\right)_{u_1,v_1}$, $(0,0)_{r_2,s_2}$ and $\left(-\frac{(\alpha-1)^5}{2\alpha^4},0\right)_{r_3,s_3}$, and the procedure returns a curve N_t associated to this vector field.

```
> r4devlim1:=proc(M,p,q)
> local prod,res,res1,res2,resu1,resu2,N,MMM:
> MMM:=(r,s)->(op(M(r,s))):
> prod:=unapply([(eeeeta2@(A@(ppppsi1@MMM@eeeeta1@(B@pppppsi2))))(r,s)],t):
> res:=simplify(subs(t=0,diff(prod(t),t))):
> res1:=unapply(simplify(op(1,res)),(r,s)):
> res2:=unapply(simplify(op(2,res)),(r,s)):
> resu1:=simplify(mtaylor(res1(r,s),[r=p,s=q],1)):
> resu2:=simplify(mtaylor(res2(r,s),[r=p,s=q],1)):
> N:=unapply([r+t*resu1,s+t*resu2],(r,s)):
> RETURN(N):
> end:
```

The procedure `r4devlim2` associates to a vector field Z on the blow up of $\mathbb{P}^2$ at $\left(-\frac{1}{\alpha},\frac{1}{\alpha}\right)_{y,z}$, $\left(0,\frac{1}{\alpha-1}\right)_{u_1,v_1}$, $(0,0)_{r_2,s_2}$ and $\left(-\frac{(\alpha-1)^5}{2\alpha^4},0\right)_{r_3,s_3}$ expressed in the coordinates (r_4,s_4) the Taylor expansion of $(A\phi)_*\,Z$ of order 0 at a point (p,q). The result is viewed as a polynomial vector field in a neighborhood of (p,q) in the coordinates (r_4,s_4) which are this time the coordinates on $\mathbb{P}^2$ blown up at $\left(\frac{1}{\alpha},\frac{1}{\alpha}\right)_{y,z}$, $\left(0,\frac{1}{\alpha+1}\right)_{u_1,v_1}$, $(0,0)_{r_2,s_2}$ and $\left(\frac{(\alpha+1)^5}{2\alpha^4},0\right)$, and the procedure returns a curve N_t associated to this vector field.

```
> r4devlim2:=proc(M,p,q)
> local prod,res,res1,res2,resu1,resu2,N,MMM:
> MMM:=(r,s)->(op(M(r,s))):
> prod:=unapply([(eeeeta3@(A@f@(ppppsi2@MMM@eeeeta2@g@B@pppppsi3)))(r,s)],t):
> res:=simplify(subs(t=0,diff(prod(t),t))):
> res1:=unapply(simplify(op(1,res)),(r,s)):
> res2:=unapply(simplify(op(2,res)),(r,s)):
> resu1:=simplify(mtaylor(res1(r,s),[r=p,s=q],1)):
> resu2:=simplify(mtaylor(res2(r,s),[r=p,s=q],1)):
> N:=unapply([r+t*resu1,s+t*resu2],(r,s)):
> RETURN(N):
> end:
```

The procedure `r4devlim3` associates to a vector field Z on the blow up of $\mathbb{P}^2$ at $\left(\frac{1}{\alpha},\frac{1}{\alpha}\right)_{y,z}$, $\left(0,\frac{1}{\alpha+1}\right)_{u_1,v_1}$, $(0,0)_{r_2,s_2}$ and $\left(\frac{(\alpha+1)^5}{2\alpha^4},0\right)$, and expressed in coordinates (r_4,s_4) the Taylor expansion of $(A\phi)_*\,Z$ of order 0 at a point (p,q). The result is viewed as a polynomial vector field in a neighborhood of (p,q) in the coordinates (r_4,s_4) which are this time the coordinates on $\mathbb{P}^2$ blown up at $(0,0)_{y,z}$, $(0,0)_{u_1,v_1}$, $(0,0)_{r_2,s_2}$ and $(-1,0)_{r_3,s_3}$, and the procedure returns a curve N_t associated to this vector field.

```
> r4devlim3:=proc(M,p,q)
> local prod,res,res1,res2,resu1,resu2,N,MMM:
> MMM:=(r,s)->(op(M(r,s))):
> prod:=unapply([(eeeeta4@(A@f@(ppppsi3@MMM@eeeeta3@g@B@pppppsi4)))(r,s)],t):
> res:=simplify(subs(t=0,diff(prod(t),t))):
> res1:=unapply(simplify(op(1,res)),(r,s)):
> res2:=unapply(simplify(op(2,res)),(r,s)):
> resu1:=simplify(mtaylor(res1(r,s),[r=p,s=q],1)):
> resu2:=simplify(mtaylor(res2(r,s),[r=p,s=q],1)):
> N:=unapply([r+t*resu1,s+t*resu2],(r,s)):
> RETURN(N):
> end:
```

The procedure `r4extract` associates to a vector field given by a curve $N_t(r_4,s_4)$ the components of ∂_{r_4} and ∂_{s_4} in the Taylor expansion of the vector field associated at $(0,0)$.

```
> r4extract:=proc(M)
> local prod,res,res1,res2,w:
> prod:=unapply(M(r,s),t):
> res:=simplify(subs(t=0,diff(prod(t),t))):
> res1:=unapply(op(1,res),(r,s)):
> res2:=unapply(op(2,res),(r,s)):
> w[1]:=simplify(res1(0,0)):
> w[2]:=simplify(res2(0,0)):
> w:=[w[1],w[2]]:
> RETURN(convert(w,Vector)):
> end:
```

1.3 Definition of the residues matrices

$\mathtt{R}_1$, denoted by $\mathscr{Z}_1$ in the article, is the (10×7)-matrix of ∂_y, $y\,\partial_y$, $z\,\partial_y$, ∂_z,$y\,\partial_z$, $z\,\partial_z$, $y^2\,\partial_z$ in the basis $(\mathfrak{l}_i)_{1\le i\le 10}$.

```
> R_1:=Transpose(Matrix([[-lambda[1],-1,2,0,0,0,-lambda[4]^2,-2*lambda[4],0,0],[0,0,0,0,0,
0,-3,0,0,0],[0,0,0,0,0,0,0,0,-3,0],[1,0,0,0,0,0,lambda[4]^3,3*lambda[4]^2,2*lambda[5]^2,4*
lambda[5]],[0,0,-lambda[2],-1,-2*lambda[3],-2,-3*lambda[4],-3,-4*lambda[5],-4],[0,0,0,0,0,
0,2,0,0,0],[0,0,0,0,0,0,0,0,0,-2,0]]));
```

$$R\_1 := \begin{bmatrix} -\lambda_1 & 0 & 0 & 1 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 2 & 0 & 0 & 0 & -\lambda_2 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -2\,\lambda_3 & 0 & 0 \\ 0 & 0 & 0 & 0 & -2 & 0 & 0 \\ -\lambda_4{}^2 & -3 & 0 & \lambda_4{}^3 & -3\,\lambda_4 & 2 & 0 \\ -2\,\lambda_4 & 0 & 0 & 3\,\lambda_4{}^2 & -3 & 0 & 0 \\ 0 & 0 & -3 & 2\,\lambda_5{}^2 & -4\,\lambda_5 & 0 & -2 \\ 0 & 0 & 0 & 4\,\lambda_5 & -4 & 0 & 0 \end{bmatrix}$$

$\mathtt{R}_2$, denoted by $\mathscr{Z}_2$ in the article, is the (10×7)-matrix of ∂_y, $y\,\partial_y$, $z\,\partial_y$, ∂_z,$y\,\partial_z$, $z\,\partial_z$, $y^2\,\partial_z$ in the basis $(\mathfrak{m}_i)_{1\le i\le 10}$.

```
> R_2:=Transpose(Matrix([[-mu[1],-1,2,0,0,0,-mu[4]^2,-2*mu[4],0,0],[0,0,0,0,0,0,3,0,0,0],
[0,0,0,0,0,0,0,0,3,0],[1,0,0,0,0,0,-mu[4]^3,-3*mu[4]^2,2*mu[5]^2,4*mu[5]],[0,0,-mu[2],-1,-2*
mu[3],-2,-3*mu[4],-3,-4*mu[5],-4],[0,0,0,0,0,0,-2,0,0,0],[0,0,0,0,0,0,0,0,0,-2,0]]));
```

$$R\_2 := \begin{bmatrix} -\mu_1 & 0 & 0 & 1 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 2 & 0 & 0 & 0 & -\mu_2 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -2\,\mu_3 & 0 & 0 \\ 0 & 0 & 0 & 0 & -2 & 0 & 0 \\ -\mu_4{}^2 & 3 & 0 & -\mu_4{}^3 & -3\,\mu_4 & -2 & 0 \\ -2\,\mu_4 & 0 & 0 & -3\,\mu_4{}^2 & -3 & 0 & 0 \\ 0 & 0 & 3 & 2\,\mu_5{}^2 & -4\,\mu_5 & 0 & -2 \\ 0 & 0 & 0 & 4\,\mu_5 & -4 & 0 & 0 \end{bmatrix}$$

$\mathtt{RR}_1$ is the (10×10)-matrix that gives the residues of ∂_{u_1}, $u_1\,\partial_{u_1}$, $v_1\,\partial_{u_1}$, $u_1v_1\,\partial_{u_1}$, $v_1^2\,\partial_{u_1}$, $v_1^3\,\partial_{u_1}$, ∂_{v_1}, $u_1\,\partial_{v_1}$, $v_1\,\partial_{v_1}$, $v_1^2\,\partial_{v_1}$ in the basis $(\mathfrak{l}_i)_{1\le i\le 10}$.

```
> RR_1:=Matrix([[0,0,0,0,0,0,0,0,0,0],[0,0,0,0,0,0,0,0,0,0],[1,0,0,0,0,0,-lambda[2],0,0,0],
[0,0,0,0,0,0,-1,0,0,0],[0,0,1,0,0,0,-2*lambda[3],0,0,0],[0,0,0,0,0,0,-2,0,0,0],[0,-1,0,0,1,
0,-3*lambda[4],0,2,0],[0,0,0,0,0,0,-3,0,0,0],[0,0,0,-1,0,1,-4*lambda[5],-2,0,2],[0,0,0,0,0,
0,-4,0,0,0]]);
```

$$RR\_1 := \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & -\lambda_2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & -2\,\lambda_3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -2 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 1 & 0 & -3\,\lambda_4 & 0 & 2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -3 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 1 & -4\,\lambda_5 & -2 & 0 & 2 \\ 0 & 0 & 0 & 0 & 0 & 0 & -4 & 0 & 0 & 0 \end{bmatrix}$$

$\mathtt{RRR}_1$ is the (10×12)-matrix that gives the residues of ∂_{r_2}, $r_2\,\partial_{r_2}$, $s_2\,\partial_{r_2}$, $r_2 s_2\,\partial_{r_2}$, $r_2^2\partial_{r_2}$, $s_2^2\,\partial_{r_2}$, ∂_{s_2}, $r_2\,\partial_{s_2}$, $s_2\,\partial_{s_2}$, $r_2 s_2\,\partial_{s_2}$, $r_2^2\,\partial_{s_2}$, $s_2^2\,\partial_{s_2}$ in the basis $(\mathfrak{l}_i)_{1\le i\le 10}$.

```
> RRR_1:=Matrix([[0,0,0,0,0,0,0,0,0,0,0,0],[0,0,0,0,0,0,0,0,0,0,0,0],[0,0,0,0,0,0,0,0,0,0,
0,0],[0,0,0,0,0,0,0,0,0,0,0,0],[1,0,0,0,0,0,-lambda[3],0,0,0,0,0],[0,0,0,0,0,0,-1,0,0,0,0,
0],[0,-1,1,0,0,0,-2*lambda[4],-1,1,0,0,0],[0,0,0,0,0,0,-2,0,0,0,0,0],[0,0,0,-1,1,1,-3*lambda[5],
0,0,-1,1,1],[0,0,0,0,0,0,-3,0,0,0,0,0]]);
```

$$RRR\_1 := \begin{bmatrix} \text{`10 x 12`}\,(Matrix) \\ \text{`Data Type:`}\;\; \text{`anything} \\ \text{`Storage:`}\;\; \text{`rectangular} \\ \text{`Order:`}\;\; \text{`Fortran\_order} \end{bmatrix}$$

```
> evalm(RRR_1);
```

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & -\lambda_3 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 & 0 & -2\,\lambda_4 & -1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 1 & 1 & -3\,\lambda_5 & 0 & 0 & -1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & -3 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$\mathtt{RRRR}_1$ is the (10×3)-matrix that gives the residues of ∂_{r_3}, $s_3\partial_{r_3}$, ∂_{s_3} in the basis $(\mathfrak{l}_i)_{1\le i\le 10}$.

```
> RRRR_1:=Matrix([[0,0,0],[0,0,0],[0,0,0],[0,0,0],[0,0,0],[0,0,0],[1,0,-lambda[4]],[0,0,-1],
[0,1,-2*lambda[5]],[0,0,-2]]);
```

$$RRRR\_1 := \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & -\lambda_4 \\ 0 & 0 & -1 \\ 0 & 1 & -2\,\lambda_5 \\ 0 & 0 & -2 \end{bmatrix}$$

RRRRR$_1$ is the (10×2)-matrix that gives the residues of ∂_{r_4}, ∂_{s_4} in the basis $(\mathfrak{l}_i)_{1 \le i \le 10}$.

```
> RRRRR_1:=Matrix([[0,0],[0,0],[0,0],[0,0],[0,0],[0,0],[0,0],[0,0],[1,-lambda[5]],[0,-1]])
;
```

$$RRRRR\_1 := \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & -\lambda_5 \\ 0 & -1 \end{bmatrix}$$

P is the transition matrix from the algebraic basis $(\mathfrak{m}_i)_{1 \le 10}$ to the geometric basis for $\widehat{P}_2$.

```
> P:=Matrix([[-mu[1],1,0,0,0,0,0,0,0,0],[-1,0,0,0,0,0,0,0,0,0],[2,0,1,-mu[2],0,0,0,0,0,0],
[0,0,0,-1,0,0,0,0,0,0],[0,0,0,-2*mu[3],1,-mu[3],0,0,0,0],[0,0,0,-2,0,-1,0,0,0,0],[-mu[4]^2,
-mu[4]^3,0,-3*mu[4],0,-2*mu[4],1,-mu[4],0,0],[-2*mu[4],-3*mu[4]^2,0,-3,0,-2,0,-1,0,0],[0,2*
mu[5]^2,0,-4*mu[5],0,-3*mu[5],0,-2*mu[5],1,-mu[5]],[0,4*mu[5],0,-4,0,-3,0,-2,0,-1]]);
```

$$P := \begin{bmatrix} -\mu_1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 2 & 0 & 1 & -\mu_2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -2\,\mu_3 & 1 & -\mu_3 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -2 & 0 & -1 & 0 & 0 & 0 & 0 \\ -\mu_4{}^2 & -\mu_4{}^3 & 0 & -3\,\mu_4 & 0 & -2\,\mu_4 & 1 & -\mu_4 & 0 & 0 \\ -2\,\mu_4 & -3\,\mu_4{}^2 & 0 & -3 & 0 & -2 & 0 & -1 & 0 & 0 \\ 0 & 2\,\mu_5{}^2 & 0 & -4\,\mu_5 & 0 & -3\,\mu_5 & 0 & -2\,\mu_5 & 1 & -\mu_5 \\ 0 & 4\,\mu_5 & 0 & -4 & 0 & -3 & 0 & -2 & 0 & -1 \end{bmatrix}$$

PP, denoted by $\mathscr{K}$ in the article, is the transition matrix from the algebraic basis $(\mathfrak{l}_i)_{1 \le i \le 10}$ to the geometric basis $(\mathfrak{e}_i)_{1 \le i \le 10}$ for $\widehat{P}_1$.

```
> PP:=Matrix([[-lambda[1],1,0,0,0,0,0,0,0,0],[-1,0,0,0,0,0,0,0,0,0],[2,0,1,-lambda[2],0,0,
0,0,0,0],[0,0,0,-1,0,0,0,0,0,0],[0,0,0,-2*lambda[3],1,-lambda[3],0,0,0,0],[0,0,0,-2,0,-1,0,
0,0,0],[-lambda[4]^2, lambda[4]^3,0,-3*lambda[4],0,-2*lambda[4],1,-lambda[4],0,0],[-2*lambda[4],
3*lambda[4]^2,0,-3,0,-2,0,-1,0,0],[0,2*lambda[5]^2,0,-4*lambda[5],0,-3*lambda[5],0,-2*lambda[5],
1,-lambda[5]],[0,4*lambda[5],0,-4,0,-3,0,-2,0,-1]]);
```

$$PP := \begin{bmatrix} -\lambda_1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 2 & 0 & 1 & -\lambda_2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -2\,\lambda_3 & 1 & -\lambda_3 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -2 & 0 & -1 & 0 & 0 & 0 & 0 \\ -\lambda_4{}^2 & \lambda_4{}^3 & 0 & -3\,\lambda_4 & 0 & -2\,\lambda_4 & 1 & -\lambda_4 & 0 & 0 \\ -2\,\lambda_4 & 3\,\lambda_4{}^2 & 0 & -3 & 0 & -2 & 0 & -1 & 0 & 0 \\ 0 & 2\,\lambda_5{}^2 & 0 & -4\,\lambda_5 & 0 & -3\,\lambda_5 & 0 & -2\,\lambda_5 & 1 & -\lambda_5 \\ 0 & 4\,\lambda_5 & 0 & -4 & 0 & -3 & 0 & -2 & 0 & -1 \end{bmatrix}$$

PPP is the transition matrix from the algebraic basis $(\mathfrak{l}_i)_{1\le i\le 10}$ to the geometric basis $(\mathfrak{e}_i)_{1\le i\le 10}$ for $\widehat{P}_1$.

```
> PPP:=MatrixInverse(PP):
```

1.4 Construction of the matrix $\mathscr{Y}$

1.4.1 Construction of a block matrix

We input the coefficients of the matrix $\mathscr{L}$.

```
> col[1]:=[-mu[1], -1, 2, 0, 0, 0, 2*mu[4]^2, 4*mu[4], 0, 0, 0, 0, 0, 0, 0, 0, -4, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, -3, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 3, 0];
```

$col_1 := [-\mu_1,-1,2,0,0,0,2\,\mu_4^2,4\,\mu_4,0,0,0,0,0,0,0,0,-4,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,-3,0,0,$
$0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,3,0]$

```
> col[2]:=
> [1, 0, 0, 0, 0, 0, -mu[4]^3, -3*mu[4]^2, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 2, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, -2];
```

$col_2 := [1,0,0,0,0,0,-\mu_4^3,-3\,\mu_4^2,0,0,0,0,0,0,0,0,0,0,0,0,0,2,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,$
$0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,-2]$

```
> col[3]:=
> [0, 0, 1, 0, 0, 0, mu[4]^2, 2*mu[4], 0, 0, 0, 0, 0, 0, 0, 0, -2, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, -2, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 1, 0];
```

$col_3 := [0,0,1,0,0,0,\mu_4^2,2\,\mu_4,0,0,0,0,0,0,0,0,-2,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,-2,0,0,0,0,0,$
$0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0]$

```
> col[4]:=
> [0, 0, -mu[2], -1, 2*mu[3]^2-2*mu[3], 4*mu[3]-2, mu[4], 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 2, 0, 0];
```

$col_4 := [0,0,-\mu_2,-1,2\,\mu_3^2-2\,\mu_3,4\,\mu_3-2,\mu_4,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,$
$0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,2,0,0]$

```
> col[5]:=
> [0, 0, -2*mu[2], -2, (mu[3]-1)^2, 2*(mu[3]-1), 0, 0, 0, 0, 0, 0, -2, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0];
```

$col_5 := [0,0,-2\,\mu_2,-2,(\mu_3-1)^2,2\,\mu_3-2,0,0,0,0,0,0,-2,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,$
$0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0]$

```
> col[6]:=
> [0, 0, 0, 0, mu[3]^2-mu[3], 2*mu[3]-1, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0];
```

$col_6 := [0,0,0,0,\mu_3^2-\mu_3,2\,\mu_3-1,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,$
$0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0]$

```
> col[7]:=
> [0, 0, mu[2]^2, 2*mu[2], 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 1, 0, 0, 0];
```

$col_7 := [0,0,\mu_2^2,2\,\mu_2,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,$
$0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0]$

```
> col[8]:=
> [0, 0, mu[2], 1, 0, 0, 0, 0, 0, 0, 0, 0, 2, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0];
```

$col_8 := [0,0,\mu_2,1,0,0,0,0,0,0,0,0,2,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0$
$,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]$

```
> col[9]:=[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 1, 0, 0, 0, 0];
```

$col_9 := [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,$
$0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0]$

```
> col[10]:=
> [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0];
```

$col_{10} := [0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]$

The matrix `PART` is by definition $\mathscr{L}\mathscr{K}^{-1}$.

```
> PART:=MatrixMatrixMultiply(Transpose(Matrix([col[1],col[2],col[3],col[4],col[5],col[6],
col[7],col[8],col[9],col[10]])),PPP);
```

$$PART := \begin{bmatrix} \text{`65 x 10`}\ (Matrix) \\ \text{`Data Type:`}\ \ \text{`}anything \\ \text{`Storage:`}\ \ \text{`}rectangular \\ \text{`Order:`}\ \ \text{`}Fortran\_order \end{bmatrix}$$

```
> for k from 1 to 85 do
> for j from 1 to 30 do
> L[k][j]:=0
> od
> od:
```

We construct the matrix $\mathscr{Y}$, whose upper right block $\mathscr{Q}$ is indeterminate.

```
> for k from 1 to 10 do
> for l from 1 to 10 do
> L[k][l+20]:=Q[k,l]
> od
> od:
> for k from 76 to 85 do
> L[k][k-65]:=1
> od:
> for k from 11 to 75 do
> for j from 1 to 10 do
> L[k][j]:=PART[k-10,j]
> od
> od:
> for k from 1 to 85 do
> L[k]:=convert(L[k],list)
> od:
> MM:=Matrix([seq(L[k],k=1..85)]):
```

1.4.2 Computation of $\mathscr{Q}$

To compute the coefficients of $\mathscr{Q}$, we consider the matrix `R` obtained by multiplying $\mathscr{Q}$ at right by the transition matrix from the algebraic basis to the geometric basis of $\widehat{P}_2$. The matrix `R` is then the matrix

$$(A\phi A\phi A)_* : W(\widehat{P}_2, \mathscr{B}_{\mathtt{geom}}) \longrightarrow W(\widehat{P}_1, \mathscr{B}_{\mathtt{alg}}).$$

Computation of $(A\phi A\phi A)_*\,\partial_y$ in the basis $(\mathfrak{l}_i)_{1\leq i\leq 10}$.

```
> y:='y': z:='z':
> M:=(y,z)->[y+t,z]:
> U:=MatrixMatrixMultiply(R_1,extract(devlim2(devlim2((devlim1(M,-1/alpha,1/alpha)), 1/alpha,
1/alpha),0,0))):
> for k from 1 to 10 do
> R[k][1]:=U[k];
> od:
```

Computation of $(A\phi A\phi A)_*\,\partial_z$ in the basis $(\mathfrak{l}_i)_{1\leq i\leq 10}$.

```
> y:='y': z:='z':
> M:=(y,z)->[y,z+t]:
> U:=MatrixMatrixMultiply(R_1,extract(devlim2(devlim2((devlim1(M,-1/alpha,1/alpha)), 1/alpha,
1/alpha),0,0))):
> for k from 1 to 10 do
> R[k][2]:=U[k];
> od:
```

Computation of $(A\phi A\phi A)_*\,\partial_{u_1}$ in the basis $(\mathfrak{l}_i)_{1\leq i\leq 10}$.

```
>  u:='u':  v:='v':
>  M:=(u,v)->[u+t,v]:
>  U:=MatrixMatrixMultiply(RR_1,u1extract(u1devlim3(u1devlim2(u1devlim1(M,0,1/(alpha-1)), 0,
1/(1+alpha)),0,0))):
>  for k from 1 to 10 do
>  R[k][3]:=U[k];
>  od:
```

Computation of $(A\phi A\phi A)_* \, \partial_{v_1}$ in the basis $(\mathfrak{l}_i)_{1\leq i\leq 10}$.

```
>  u:='u':  v:='v':
>  M:=(u,v)->[u,v+t]:
>  U:=MatrixMatrixMultiply(RR_1,u1extract(u1devlim3(u1devlim2(u1devlim1(M,0,1/(alpha-1)), 0,
1/(1+alpha)),0,0))):
>  for k from 1 to 10 do
>  R[k][4]:=U[k];
>  od:
```

Computation of $(A\phi A\phi A)_* \, \partial_{r_2}$ in the basis $(\mathfrak{l}_i)_{1\leq i\leq 10}$.

```
>  r:='r':  s:='s':
>  M:=(r,s)->[r+t,s]:
>  U:=MatrixMatrixMultiply(RRR_1,r2extract(r2devlim3(r2devlim2(r2devlim1(M,0,0),0,0),0,0))
):
>  for k from 1 to 10 do
>  R[k][5]:=U[k];
>  od:
```

Computation of $(A\phi A\phi A)_* \, \partial_{s_2}$ in the basis $(\mathfrak{l}_i)_{1\leq i\leq 10}$.

```
>  r:='r':  s:='s':
>  M:=(r,s)->[r,s+t]:
>  U:=MatrixMatrixMultiply(RRR_1,r2extract(r2devlim3(r2devlim2(r2devlim1(M,0,0),0,0),0,0))
):
>  for k from 1 to 10 do
>  R[k][6]:=U[k];
>  od:
```

Computation of $(A\phi A\phi A)_* \, \partial_{r_3}$ in the basis $(\mathfrak{l}_i)_{1\leq i\leq 10}$.

```
>  r:='r':  s:='s':
>  M:=(r,s)->[r+t,s]:
>  U:=MatrixMatrixMultiply(RRRR_1,r3extract(r3devlim3(r3devlim2(r3devlim1(M,-(1/2)*(-1+alpha)
^5/alpha^4,0),(1/2)*(1+alpha)^5/alpha^4, 0),-1,0))):
>  for k from 1 to 10 do
>  R[k][7]:=U[k];
>  od:
```

Computation of $(A\phi A\phi A)_* \, \partial_{s_3}$ in the basis $(\mathfrak{l}_i)_{1\leq i\leq 10}$.

```
>  r:='r':  s:='s':
>  M:=(r,s)->[r,s+t]:
>  U:=MatrixMatrixMultiply(RRRR_1,r3extract(r3devlim3(r3devlim2(r3devlim1(M,-(1/2)*(-1+alpha)
^5/alpha^4,0),(1/2)*(1+alpha)^5/alpha^4, 0),-1,0))):
>  for k from 1 to 10 do
>  R[k][8]:=U[k];
>  od:
```

Computation of $(A\phi A\phi A)_* \, \partial_{c_4}$ in the basis $(\mathfrak{l}_i)_{1\leq i\leq 10}$.

```
>  r:='r':  s:='s':
>  M:=(r,s)->[r+t,s]:
>  U:=MatrixMatrixMultiply(RRRRR_1,r4extract(r4devlim3(r4devlim2(r4devlim1(M,(3/4)*(-1+alpha)
^4*(alpha^3-alpha-alpha^2+1)/alpha^5,0),-(3/4)*(1+alpha)^4*(-alpha-1+alpha^3+alpha^2)/alpha^5,
0),0,0))):
>  for k from 1 to 10 do
>  R[k][9]:=U[k];
>  od:
```

Computation of $(A\phi A\phi A)_* \, \partial_{d_4}$ in the basis $(\mathfrak{l}_i)_{1\leq i\leq 10}$.

```
>  r:='r':  s:='s':
>  M:=(r,s)->[r,s+t]:
>  U:=MatrixMatrixMultiply(RRRRR_1,r4extract(r4devlim3(r4devlim2(r4devlim1(M,(3/4)*(-1+alpha)
^4*(alpha^3-alpha-alpha^2+1)/alpha^5,0),-(3/4)*(1+alpha)^4*(-alpha-1+alpha^3+alpha^2)/alpha^5,
0),0,0))):
>  for k from 1 to 10 do
>  R[k][10]:=U[k];
>  od:
```

We construct the matrix `R`, then we compute $\mathscr{Q}$.

```
>  for k from 1 to 10 do
```

```
> R[k]:=convert(R[k],list)
> od:
> TEMP:=Matrix([seq(R[k],k=1..10)]):
> Q:=MatrixMatrixMultiply(TEMP, MatrixInverse(P)):
```

We can display the coefficients of $\mathscr{Q}$.

```
> simplify(Q[9,4]);
```

$$4\,(23+\alpha-6\,\lambda_5)\mu_2$$

1.5 The 8 vectors of $V(\mathfrak{D}_1)$ that form the columns of the matrix $\mathscr{V}_1$

1.5.1 Columns of $\mathscr{M}_a$ (eight (10×1)-vectors)

a_1

```
> pro1:=[-lambda[1],-1,2,0,0,0,-lambda[4]^2,-2*lambda[4],0,0];
> a[1]:=convert(pro1,Vector):
```

$$pro1 := [-\lambda_1, -1, 2, 0, 0, 0, -{\lambda_4}^2, -2\,\lambda_4, 0, 0]$$

a_2

```
> pro2:=[0,0,0,0,0,0,-3,0,0,0];
> a[2]:=convert(pro2,Vector):
```

$$pro2 := [0, 0, 0, 0, 0, 0, -3, 0, 0, 0]$$

a_3

```
> pro3:=[0,0,0,0,0,0,0,0,-3,0];
> a[3]:=convert(pro3,Vector):
```

$$pro3 := [0, 0, 0, 0, 0, 0, 0, 0, -3, 0]$$

a_4

```
> pro4:=[1,0,0,0,0,0,lambda[4]^3,3*lambda[4]^2,2*lambda[5]^2,4*lambda[5]];
> a[4]:=convert(pro4,Vector):
```

$$pro4 := [1, 0, 0, 0, 0, 0, {\lambda_4}^3, 3\,{\lambda_4}^2, 2\,{\lambda_5}^2, 4\,\lambda_5]$$

a_5

```
> pro5:=[0,0,-lambda[2],-1,-2*lambda[3],-2,-3*lambda[4],-3,-4*lambda[5],-4];
> a[5]:=convert(pro5,Vector):
```

$$pro5 := [0, 0, -\lambda_2, -1, -2\,\lambda_3, -2, -3\,\lambda_4, -3, -4\,\lambda_5, -4]$$

a_6

```
> pro6:=[0,0,0,0,0,0,2,0,0,0];
> a[6]:=convert(pro6,Vector):
```

$$pro6 := [0, 0, 0, 0, 0, 0, 2, 0, 0, 0]$$

a_7

```
> pro7:=[0,0,0,0,0,0,0,0,0,0];
> a[7]:=convert(pro7,Vector):
```

$$pro7 := [0, 0, 0, 0, 0, 0, 0, 0, 0, 0]$$

a_8

```
> pro8:=[0,0,0,0,0,0,0,0,0,0];
> a[8]:=convert(pro8,Vector):
```

$$pro8 := [0, 0, 0, 0, 0, 0, 0, 0, 0, 0]$$

1.5.2 Columns of $\mathscr{M}'_a$ (eight (10×1)-vectors)

a_1

```
> M:=(y,z)->(t+y,z):
> aa[1]:=MatrixMatrixMultiply(R_2,champ4(M)):
```

a_2

```
> M:=(y,z)->((1+t)*y,z):
> aa[2]:=MatrixMatrixMultiply(R_2,champ4(M)):
```

a_3

```
> M:=(y,z)->(y+t*z,z):
> aa[3]:=MatrixMatrixMultiply(R_2,champ4(M)):
```

a_4

```
> M:=(y,z)->(y,t+z):
> aa[4]:=MatrixMatrixMultiply(R_2,champ4(M)):
```

a_5

```
> M:=(y,z)->(y,t*y+z):
> aa[5]:=MatrixMatrixMultiply(R_2,champ4(M)):
```

a_6

```
> M:=(y,z)->(y,(1+t)*z):
> aa[6]:=MatrixMatrixMultiply(R_2,champ4(M)):
```

a_7

```
> M:=(y,z)->(y/(1-t*y),z/(1-t*y)):
> aa[7]:=MatrixMatrixMultiply(R_2,champ4(M)):
```

a_8

```
> M:=(y,z)->(y/(1-t*z),z/(1-t*z)):
> aa[8]:=MatrixMatrixMultiply(R_2,champ4(M)):
```

1.5.3 Columns of $\mathscr{M}''_a$ (eight (10×1)-vectors)

a_1

```
> M:=(y,z)->(t+y,z):
> aaa[1]:=MatrixMatrixMultiply(R_2,champ3(M)):
```

a_2

```
> M:=(y,z)->((1+t)*y,z):
> aaa[2]:=MatrixMatrixMultiply(R_2,champ3(M)):
```

a_3

```
> M:=(y,z)->(y+t*z,z):
> aaa[3]:=MatrixMatrixMultiply(R_2,champ3(M)):
```

a_4

```
> M:=(y,z)->(y,t+z):
> aaa[4]:=MatrixMatrixMultiply(R_2,champ3(M)):
```

a_5

```
> M:=(y,z)->(y,t*y+z):
> aaa[5]:=MatrixMatrixMultiply(R_2,champ3(M)):
```

a_6

```
> M:=(y,z)->(y,(1+t)*z):
> aaa[6]:=MatrixMatrixMultiply(R_2,champ3(M)):
```

a_7

```
> M:=(y,z)->(y/(1-t*y),z/(1-t*y)):
> aaa[7]:=MatrixMatrixMultiply(R_2,champ3(M)):
```

a_8

```
> M:=(y,z)->(y/(1-t*z),z/(1-t*z)):
> aaa[8]:=MatrixMatrixMultiply(R_2,champ3(M)):
```

1.5.4 Construction of the matrix $\mathscr{V}_1$

```
> mise:=proc(k::integer)
> local u,v,w,h:
> u:=convert(Transpose(a[k]),list):
> v:=convert(Transpose(aa[k]),list):
> w:=convert(Transpose(aaa[k]),list):
> h:=[op(u),op(v),op(w)]:
> RETURN(convert(h,Vector))
> end:
> for k from 1 to 8 do
> VectDef[k]:=mise(k)
> od:
```

1.6 The 63 vectors of $V(\mathfrak{D}_2)$ that form the columns of the matrix $\mathscr{V}_2$

1.6.1 Vectors of type a (we compute the columns of the matrices $\mathscr{N}_{\mathrm{a}}'$, $\mathscr{N}_{\mathrm{a}}$ and $\mathscr{N}_{\mathrm{a}}''$)

Columns of $\mathscr{N}_{\mathrm{a}}'$ *(eight* (10×1)*-vectors)*

Aa_1

```
> M:=(y,z)->(t+y,z):
> aA[1]:=MatrixMatrixMultiply(R_1,champ1(M)):
```

Aa_2

```
> M:=(y,z)->((1+t)*y,z):
> aA[2]:=MatrixMatrixMultiply(R_1,champ1(M)):
```

Aa_3

```
> M:=(y,z)->(y+t*z,z):
> aA[3]:=MatrixMatrixMultiply(R_1,champ1(M)):
```

Aa_4

```
> M:=(y,z)->(y,t+z):
> aA[4]:=MatrixMatrixMultiply(R_1,champ1(M)):
```

Aa_5

```
> M:=(y,z)->(y,t*y+z):
> aA[5]:=MatrixMatrixMultiply(R_1,champ1(M)):
```

Aa_6

```
> M:=(y,z)->(y,(1+t)*z):
> aA[6]:=MatrixMatrixMultiply(R_1,champ1(M)):
```

Aa_7

```
> M:=(y,z)->(y/(1-t*y),z/(1-t*y)):
> aA[7]:=MatrixMatrixMultiply(R_1,champ1(M)):
```

Aa_8

```
> M:=(y,z)->(y/(1-t*z),z/(1-t*z)):
> aA[8]:=MatrixMatrixMultiply(R_1,champ1(M)):
```

Columns of $\mathscr{N}_{\mathrm{a}}$ *to which we add* 55 *zeros*

Aa_1

```
> pro1:=[-mu[1],-1,2,0,0,0,-mu[4]^2,-2*mu[4],0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0];
> nops(pro1);
> aaA[1]:=convert(pro1,Vector):
```

$$pro1 := [-\mu_1, -1, 2, 0, 0, 0, -\mu_4^2, -2\,\mu_4, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]$$

$$65$$

Aa_2

```
> pro2:=[0,0,0,0,0,0,3,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0];
> nops(pro2);
> aaA[2]:=convert(pro2,Vector):
```

$\mathit{pro2} := [0,0,0,0,0,0,3,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0$
$,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]$

65

Aa_3

```
> pro3:=[0,0,0,0,0,0,0,0,3,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0];
> nops(pro3);
> aaA[3]:=convert(pro3,Vector):
```

$\mathit{pro3} := [0,0,0,0,0,0,0,0,3,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0$
$,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]$

65

Aa_4

```
> pro4:=[1,0,0,0,0,0,-mu[4]^3,-3*mu[4]^2,2*mu[5]^2,4*mu[5],0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0];
> nops(pro4);
> aaA[4]:=convert(pro4,Vector):
```

$\mathit{pro4} := [1,0,0,0,0,0,-\mu_4^3,-3\,\mu_4^2,2\,\mu_5^2,4\,\mu_5,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,$
$0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]$

65

Aa_5

```
> pro5:=[0,0,-mu[2],-1,-2*mu[3],-2,-3*mu[4],-3,-4*mu[5],-4,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0];
> nops(pro5);
> aaA[5]:=convert(pro5,Vector):
```

$\mathit{pro5} := [0,0,-\mu_2,-1,-2\,\mu_3,-2,-3\,\mu_4,-3,-4\,\mu_5,-4,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,$
$0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]$

65

Aa_6

```
> pro6:=[0,0,0,0,0,0,-2,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0];
> nops(pro6);
> aaA[6]:=convert(pro6,Vector):
```

$\mathit{pro6} := [0,0,0,0,0,0,-2,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0$
$,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]$

65

Aa_7

```
> pro7:=[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0];
> nops(pro7);
> aaA[7]:=convert(pro7,Vector):
```

$\mathit{pro7} := [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,$
$0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]$

65

Aa_8

```
> pro8:=[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0];
> nops(pro8);
> aaA[8]:=convert(pro7,Vector):
```

$pro8 := [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]$

65

Columns of $\mathcal{N}_{\mathrm{a}}''$ (*eight* (10×1)*-vectors*)

Aa_1

```
> M:=(y,z)->(t+y,z):
> aaaA[1]:=simplify(MatrixMatrixMultiply(R_2,champ2(M))):
```

Aa_2

```
> M:=(y,z)->((1+t)*y,z):
> aaaA[2]:=simplify(MatrixMatrixMultiply(R_2,champ2(M))):
```

Aa_3

```
> M:=(y,z)->(y+t*z,z):
> aaaA[3]:=simplify(MatrixMatrixMultiply(R_2,champ2(M))):
```

Aa_4

```
> M:=(y,z)->(y,t+z):
> aaaA[4]:=simplify(MatrixMatrixMultiply(R_2,champ2(M))):
```

Aa_5

```
> M:=(y,z)->(y,t*y+z):
> aaaA[5]:=simplify(MatrixMatrixMultiply(R_2,champ2(M))):
```

Aa_6

```
> M:=(y,z)->(y,(1+t)*z):
> aaaA[6]:=simplify(MatrixMatrixMultiply(R_2,champ2(M))):
```

Aa_7

```
> M:=(y,z)->(y/(1-t*y),z/(1-t*y)):
> aaaA[7]:=simplify(MatrixMatrixMultiply(R_2,champ2(M))):
```

Aa_8

```
> M:=(y,z)->(y/(1-t*z),z/(1-t*z)):
> aaaA[8]:=simplify(MatrixMatrixMultiply(R_2,champ2(M))):
```

Glueing

We form the first eight vectors of type a of $V(\mathfrak{D}_2)$.

```
> for k from 1 to 8 do
> for j from 1 to 10 do
> VectDef_a[k][j]:=aA[k][j]
> od
> od:
> for k from 1 to 8 do
> for j from 11 to 75 do
> VectDef_a[k][j]:=aaA[k][j-10]
> od
> od:
> for k from 1 to 8 do
> for j from 76 to 85 do
> VectDef_a[k][j]:=aaaA[k][j-75]
> od
> od:
> for k from 1 to 8 do
> VectDef_a[k]:=convert(VectDef_a[k],list)
> od:
> for k from 1 to 8 do
> VectDef_a[k]:=convert(VectDef_a[k],Vector)
> od:
```

1.6.2 Vectors of type b (we compute the columns of the matrices $\mathscr{N}_{\mathrm{b}}'$, $\mathscr{N}_{\mathrm{b}}$ and $\mathscr{N}_{\mathrm{b}}''$)

Columns of $\mathscr{N}_{\mathrm{b}}'$ (twenty five (10×1)-vectors)

```
> Mg:=(y,z,p,q)->(y+t*y^p/z^q,z);
```

$$Mg := (y, z, p, q) \mapsto y + \frac{t y^p}{z^q}, z$$

```
> M:=(y,z)->Mg(y,z,p,q);
```

$$M := (y, z) \mapsto Mg(y, z, p, q)$$

$Ab_{p,q}$

```
> bA:=unapply(MatrixMatrixMultiply(R_1,champ1(M)),(p,q)):
```

Colums of $\mathscr{N}_{\mathrm{b}}$ to which we add a (25×25)-identity block then 30 zeros

$Ab_{0,1}$

```
> pro1:=[0,0,0,0,0,0,2*mu[4]^5,10*mu[4]^4,3*mu[5]^3,9*mu[5]^2, 1,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0];
> nops(pro1);
> bbA[0,1]:=convert(pro1,Vector):
> bbA[0,1][11];
```

$pro1 := [0, 0, 0, 0, 0, 0, 2\,\mu_4^5, 10\,\mu_4^4, 3\,\mu_5^3, 9\,\mu_5^2, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,$
$0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]$

65

1

$Ab_{1,1}$

```
> pro2:=[-1,0,0,0,0,0,mu[4]^3,3*mu[4]^2,-2*mu[5]^2,-4*mu[5],0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0];
> nops(pro2);
> bbA[1,1]:=convert(pro2,Vector):
> bbA[1,1][12];
```

$pro2 := [-1, 0, 0, 0, 0, 0, \mu_4^3, 3\,\mu_4^2, -2\,\mu_5^2, -4\,\mu_5, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,$
$0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]$

65

1

$Ab_{2,1}$

```
> pro3:=[0,0,2*mu[2],2,3*mu[3],3,4*mu[4],4,5*mu[5],5, 0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0];
> nops(pro3);
> bbA[2,1]:=convert(pro3,Vector):
> bbA[2,1][13];
```

$pro3 := [0, 0, 2\,\mu_2, 2, 3\,\mu_3, 3, 4\,\mu_4, 4, 5\,\mu_5, 5, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,$
$0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]$

65

1

$Ab_{0,2}$

```
> pro4:=[0,0,0,0,0,0,-9*mu[4]^8,-72*mu[4]^7,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0];
> nops(pro4);
> bbA[0,2]:=convert(pro4,Vector):
> bbA[0,2][14];
```

$pro4 := [0, 0, 0, 0, 0, 0, -9\,\mu_4^8, -72\,\mu_4^7, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,$
$0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]$

65

1

$Ab_{1,2}$

```
> pro5:=[0,0,0,0,0,0,-3*mu[4]^6,-18*mu[4]^5,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0];
> nops(pro5);
> bbA[1,2]:=convert(pro5,Vector):
> bbA[1,2][15];
```

$pro5 := [0,0,0,0,0,0,-3\,\mu_4^6,-18\,\mu_4^5,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,$
$0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]$

65

1

$Ab_{2,2}$

```
> pro6:=[-1/mu[1],1/mu[1]^2,0,0,0,0,-mu[4]^4,-4*mu[4]^3,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0];
> nops(pro6);
> bbA[2,2]:=convert(pro6,Vector):
> bbA[2,2][16];
```

$pro6 := [-\mu_1^{-1},\mu_1^{-2},0,0,0,0,-\mu_4^4,-4\,\mu_4^3,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,$
$0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]$

65

1

$Ab_{3,2}$

```
> pro7:=[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0];
> nops(pro7);
> bbA[3,2]:=convert(pro7,Vector):
> bbA[3,2][17];
```

$pro7 := [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,$
$0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]$

65

1

$Ab_{0,3}$

```
> pro8:=[0,0,0,0,0,0,52*mu[4]^11,572*mu[4]^10,-28*mu[5]^6,-168*mu[5]^5,0,0,0,0,0,0,0,1,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0];
> nops(pro8);
> bbA[0,3]:=convert(pro8,Vector):
> bbA[0,3][18];
```

$pro8 := [0,0,0,0,0,0,52\,\mu_4^{11},572\,\mu_4^{10},-28\,\mu_5^6,-168\,\mu_5^5,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,$
$0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]$

65

1

$Ab_{1,3}$

```
> pro9:=[0,0,0,0,0,0,15*mu[4]^9,135*mu[4]^8,12*mu[5]^5,60*mu[5]^4,0,0,0,0,0,0,0,0,1,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0];
> nops(pro1);
> bbA[1,3]:=convert(pro9,Vector):
> bbA[1,3][19];
```

$pro9 := [0,0,0,0,0,0,15\,\mu_4^9,135\,\mu_4^8,12\,\mu_5^5,60\,\mu_5^4,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,$
$0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]$

65

1

$Ab_{2,3}$

```
> pro10:=[0,0,0,0,0,0,4*mu[4]^7,28*mu[4]^6,-5*mu[5]^4,-20*mu[5]^3,0,0,0,0,0,0,0,0,0,1,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0];
> nops(pro10);
> bbA[2,3]:=convert(pro10,Vector):
> bbA[2,3][20];
```

$$pro10 := [0,0,0,0,0,0,4\,\mu_4^7,28\,\mu_4^6,-5\,\mu_5^4,-20\,\mu_5^3,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]$$

65

1

$Ab_{3,3}$

```
> pro11:=[-1/mu[1]^2,2/mu[1]^3,0,0,0,0,mu[4]^5,5*mu[4]^4,2*mu[5]^3,6*mu[5]^2,0,0,0,0,0,0,
0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0];
> nops(pro11);
> bbA[3,3]:=convert(pro11,Vector):
> bbA[3,3][21];
```

$$pro11 := [-\mu_1^{-2},2\,\mu_1^{-3},0,0,0,0,\mu_4^5,5\,\mu_4^4,2\,\mu_5^3,6\,\mu_5^2,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]$$

65

1

$Ab_{4,3}$

```
> pro12:=[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0];
> nops(pro12);
> bbA[4,3]:=convert(pro12,Vector):
> bbA[4,3][22];
```

$$pro12 := [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]$$

65

1

$Ab_{0,4}$

```
> pro13:=[0,0,0,0,0,0,-340*mu[4]^14,-4760*mu[4]^13,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0];
> nops(pro13);
> bbA[0,4]:=convert(pro13,Vector):
> bbA[0,4][23];
```

$$pro13 := [0,0,0,0,0,0,-340\,\mu_4^{14},-4760\,\mu_4^{13},0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]$$

65

1

$Ab_{1,4}$

```
> pro14:=[0,0,0,0,0,0,-91*mu[4]^12,-1092*mu[4]^11,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0];
> nops(pro14);
> bbA[1,4]:=convert(pro14,Vector):
> bbA[1,4][24];
```

$$pro14 := [0,0,0,0,0,0,-91\,\mu_4^{12},-1092\,\mu_4^{11},0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]$$

65

1

$Ab_{2,4}$

```
> pro15:=[0,0,0,0,0,0,-22*mu[4]^10,-220*mu[4]^9,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0];
> nops(pro15);
> bbA[2,4]:=convert(pro15,Vector):
> bbA[2,4][25];
```

$$pro15 := [0,0,0,0,0,0,-22\,\mu_4^{10},-220\,\mu_4^{9},0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]$$

65

1

$Ab_{3,4}$

```
> pro16:=[0,0,0,0,0,0,-5*mu[4]^8,-40*mu[4]^7,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0];
> nops(pro16);
> bbA[3,4]:=convert(pro16,Vector):
> bbA[3,4][26];
```

$$pro16 := [0,0,0,0,0,0,-5\,\mu_4^{8},-40\,\mu_4^{7},0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]$$

65

1

$Ab_{4,4}$

```
> pro17:=[-1/mu[1]^3,3/mu[1]^4,0,0,0,0,-mu[4]^6,-6*mu[4]^5,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0];
> nops(pro17);
> bbA[4,4]:=convert(pro17,Vector):
> bbA[4,4][27];
```

$$pro17 := [-\mu_1^{-3},3\,\mu_1^{-4},0,0,0,0,-\mu_4^{6},-6\,\mu_4^{5},0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]$$

65

1

$Ab_{5,4}$

```
> pro18:=[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0];
> nops(pro18);
> bbA[5,4]:=convert(pro18,Vector):
> bbA[5,4][28];
```

$$pro18 := [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]$$

65

1

$Ab_{0,5}$

```
> pro19:=[0,0,0,0,0,0,2394*mu[4]^17,40698*mu[4]^16,429*mu[5]^9,3861*mu[5]^8,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0];
> nops(pro19);
> bbA[0,5]:=convert(pro19,Vector):
> bbA[0,5][29];
```

$$pro19 := [0,0,0,0,0,0,2394\,\mu_4^{17},40698\,\mu_4^{16},429\,\mu_5^{9},3861\,\mu_5^{8},0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]$$

65

1

$Ab_{1,5}$

```
> pro20:=[0,0,0,0,0,0,612*mu[4]^15,9180*mu[4]^14,-165*mu[5]^8,-1320*mu[5]^7,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0];
> nops(pro20);
> bbA[1,5]:=convert(pro20,Vector):
> bbA[1,5][30];
```

$$pro20 := [0,0,0,0,0,0,612\,\mu_4^{15},9180\,\mu_4^{14},-165\,\mu_5^{8},-1320\,\mu_5^{7},0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]$$

65

1

$Ab_{2,5}$

```
> pro21:=[0,0,0,0,0,0,140*mu[4]^13,1820*mu[4]^12,60*mu[5]^7,420*mu[5]^6,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0];
> nops(pro21);
> bbA[2,5]:=convert(pro21,Vector):
> bbA[2,5][31];
```

$$pro21 := [0,0,0,0,0,0,140\,\mu_4^{13},1820\,\mu_4^{12},60\,\mu_5^{7},420\,\mu_5^{6},0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]$$

65

1

$Ab_{3,5}$

```
> pro22:=[0,0,0,0,0,0,30*mu[4]^11,330*mu[4]^10,-21*mu[5]^6,-126*mu[5]^5,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0];
> nops(pro22);
> bbA[3,5]:=convert(pro22,Vector):
> bbA[3,5][32];
```

$$pro22 := [0,0,0,0,0,0,30\,\mu_4^{11},330\,\mu_4^{10},-21\,\mu_5^{6},-126\,\mu_5^{5},0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]$$

65

1

$Ab_{4,5}$

```
> pro23:=[0,0,0,0,0,0,6*mu[4]^9,54*mu[4]^8,7*mu[5]^5,35*mu[5]^4,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0];
> nops(pro23);
> bbA[4,5]:=convert(pro23,Vector):
> bbA[4,5][33];
```

$$pro23 := [0,0,0,0,0,0,6\,\mu_4^{9},54\,\mu_4^{8},7\,\mu_5^{5},35\,\mu_5^{4},0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]$$

65

1

$Ab_{5,5}$

```
> pro24:=[-1/mu[1]^4,4/mu[1]^5,0,0,0,0,mu[4]^7,7*mu[4]^6,-2*mu[5]^4,-8*mu[5]^3,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0];
> nops(pro24);
> bbA[5,5]:=convert(pro24,Vector):
> bbA[5,5][34];
```

$$pro24 := [-\mu_1^{-4},4\,\mu_1^{-5},0,0,0,0,\mu_4^{7},7\,\mu_4^{6},-2\,\mu_5^{4},-8\,\mu_5^{3},0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]$$

65

1

$Ab_{6,5}$

```
> pro25:=[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0];
> nops(pro25);
> bbA[6,5]:=convert(pro25,Vector):
> bbA[6,5][35];
```

$pro25 := [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]$

65

1

Columns of $\mathscr{N}_{\mathrm{b}}''$ *(twenty five* (10×1)*-vectors)*

```
> Mg:=(y,z,p,q)->(y+t*y^p/z^q,z);
```

$Mg := (y,z,p,q) \mapsto y + \frac{ty^p}{z^q}, z$

```
> M:=(y,z)->Mg(y,z,p,q);
```

$M := (y,z) \mapsto Mg(y,z,p,q)$

$Ab_{p,q}$

```
> bbbA:=unapply(MatrixMatrixMultiply(R_2,champ2(M)),(p,q)):
```

Glueing
We form the 25 vectors of type b of $V(\mathfrak{D}_2)$.

```
> for q from 1 to 5 do
> for p from 0 to q+1 do
> for j from 1 to 10 do
> u[p,q][j]:=bA(p,q)[j]
> od
> od
> od:
> for q from 1 to 5 do
> for p from 0 to q+1 do
> for j from 11 to 75 do
> u[p,q][j]:=bbA[p,q][j-10]
> od
> od
> od:
> for q from 1 to 5 do
> for p from 0 to q+1 do
> for j from 76 to 85 do
> u[p,q][j]:=bbbA(p,q)[j-75]
> od
> od
> od:
> for q from 1 to 5 do
> for p from 0 to q+1 do
> u[p,q]:=convert(u[p,q],list)
> od
> od:
> q:=1:
> for p from 0 to q+1 do
> VectDef_b[p+1]:=u[p,q]
> od:
> q:=2:
> for p from 0 to q+1 do
> VectDef_b[p+4]:=u[p,q]
> od:
> q:=3:
> for p from 0 to q+1 do
> VectDef_b[p+8]:=u[p,q]
> od:
> q:=4:
> for p from 0 to q+1 do
> VectDef_b[p+13]:=u[p,q]
```

```
> od:
> q:=5:
> for p from 0 to q+1 do
> VectDef_b[p+19]:=u[p,q]
> od:
> p:='p':
> q:='q':
```

1.6.3 Vectors of type c (we compute the columns of the matrices $\mathscr{N}_c'$, $\mathscr{N}_c$ and $\mathscr{N}_c''$)

Columns of $\mathscr{N}_c'$ *(twenty five* (10×1)*-vectors)*

```
> Mg:=(y,z,p,q)->(y,z+t*y^p/z^q);
```

$$Mg := (y, z, p, q) \mapsto y, z + \frac{t y^p}{z^q}$$

```
> M:=(y,z)->Mg(y,z,p,q);
```

$$M := (y, z) \mapsto Mg(y, z, p, q)$$

$Ac_{p,q}$

```
> cA:=unapply(MatrixMatrixMultiply(R_1,champ1(M)),(p,q)):
```

Columns of $\mathscr{N}_c$ *to which we add 25 zeros, a* (25×25)*-identity block then 5 zeros*

$Ac_{0,1}$

```
> pro1:=[0,0,0,0,0,0,4*mu[4]^6,24*mu[4]^5,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0];
> nops(pro1);
> ccA[0,1]:=convert(pro1,Vector):
> ccA[0,1][36];
```

$$pro1 := [0,0,0,0,0,0,4\,\mu_4^6,24\,\mu_4^5,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]$$

65

1

$Ac_{1,1}$

```
> pro2:=[1/mu[1],-1/mu[1]^2,0,0,0,0,mu[4]^4,4*mu[4]^3,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0];
> nops(pro2);
> ccA[1,1]:=convert(pro2,Vector):
> ccA[1,1][37];
```

$$pro2 := [\mu_1^{-1},-\mu_1^{-2},0,0,0,0,\mu_4^4,4\,\mu_4^3,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]$$

65

1

$Ac_{2,1}$

```
> pro3:=[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0];
> nops(pro3);
> ccA[2,1]:=convert(pro3,Vector):
> ccA[2,1][38];
```

$$pro3 := [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]$$

65

1

$Ac_{0,2}$

```
> pro4:=[0,0,0,0,0,0,-25*mu[4]^9,-225*mu[4]^8,-18*mu[5]^5,-90*mu[5]^4,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0];
```

```
> nops(pro4);
> ccA[0,2]:=convert(pro4,Vector):
> ccA[0,2][39];
```

$$pro4 := [0, 0, 0, 0, 0, 0, -25\,\mu_4^9, -225\,\mu_4^8, -18\,\mu_5^5, -90\,\mu_5^4, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]$$

65

1

$Ac_{1,2}$

```
> pro5:=[0,0,0,0,0,0,-5*mu[4]^7,-35*mu[4]^6,6*mu[5]^4,24*mu[5]^3,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0];
> nops(pro5);
> ccA[1,2]:=convert(pro5,Vector):
> ccA[1,2][40];
```

$$pro5 := [0, 0, 0, 0, 0, 0, -5\,\mu_4^7, -35\,\mu_4^6, 6\,\mu_5^4, 24\,\mu_5^3, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]$$

65

1

$Ac_{2,2}$

```
> pro6:=[1/mu[1]^2,-2/mu[1]^3,0,0,0,0,-mu[4]^5,-5*mu[4]^4,-2*mu[5]^3,-6*mu[5]^2,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0];
> nops(pro6);
> ccA[2,2]:=convert(pro6,Vector):
> ccA[2,2][41];
```

$$pro6 := [\mu_1^{-2}, -2\,\mu_1^{-3}, 0, 0, 0, 0, -\mu_4^5, -5\,\mu_4^4, -2\,\mu_5^3, -6\,\mu_5^2, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]$$

65

1

$Ac_{3,2}$

```
> pro7:=[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0];
> nops(pro7);
> ccA[3,2]:=convert(pro7,Vector):
> ccA[3,2][42];
```

$$pro7 := [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]$$

65

1

$Ac_{0,3}$

```
> pro8:=[0,0,0,0,0,0,182*mu[4]^12,2184*mu[4]^11,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0];
> nops(pro8);
> ccA[0,3]:=convert(pro8,Vector):
> ccA[0,3][43];
```

$$pro8 := [0, 0, 0, 0, 0, 0, 182\,\mu_4^{12}, 2184\,\mu_4^{11}, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]$$

65

1

$Ac_{1,3}$

```
> pro9:=[0,0,0,0,0,0,33*mu[4]^10,330*mu[4]^9,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0];
```

```
> nops(pro1);
> ccA[1,3]:=convert(pro9,Vector):
> ccA[1,3][44];
```

$pro9 := [0, 0, 0, 0, 0, 0, 33\,\mu_4^{10}, 330\,\mu_4^{9}, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,$
$0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]$

65

1

$Ac_{2,3}$

```
> pro10:=[0,0,0,0,0,0,6*mu[4]^8,48*mu[4]^7,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0];
> nops(pro10);
> ccA[2,3]:=convert(pro10,Vector):
> ccA[2,3][45];
```

$pro10 := [0, 0, 0, 0, 0, 0, 6\,\mu_4^{8}, 48\,\mu_4^{7}, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,$
$0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]$

65

1

$Ac_{3,3}$

```
> pro11:=[1/mu[1]^3,-3/mu[1]^4,0,0,0,0,mu[4]^6,6*mu[4]^5,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0];
> nops(pro11);
> ccA[3,3]:=convert(pro11,Vector):
> ccA[3,3][46];
```

$pro11 := [\mu_1^{-3}, -3\,\mu_1^{-4}, 0, 0, 0, 0, \mu_4^{6}, 6\,\mu_4^{5}, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,$
$0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]$

65

1

$Ac_{4,3}$

```
> pro12:=[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0];
> nops(pro12);
> ccA[4,3]:=convert(pro12,Vector):
> ccA[4,3][47];
```

$pro12 := [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,$
$1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]$

65

1

$Ac_{0,4}$

```
> pro13:=[0,0,0,0,0,0,-1428*mu[4]^15,-21420*mu[4]^14,330*mu[5]^8,2640*mu[5]^7,
> 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0];
> nops(pro13);
> ccA[0,4]:=convert(pro13,Vector):
> ccA[0,4][48];
```

$pro13 := [0, 0, 0, 0, 0, 0, -1428\,\mu_4^{15}, -21420\,\mu_4^{14}, 330\,\mu_5^{8}, 2640\,\mu_5^{7}, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,$
$0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]$

65

1

$Ac_{1,4}$

```
> pro14:=[0,0,0,0,0,0,-245*mu[4]^13,-3185*mu[4]^12,-96*mu[5]^7,-672*mu[5]^6,
> 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0];
```

```
> nops(pro14);
> ccA[1,4]:=convert(pro14,Vector):
> ccA[1,4][49];
```

$$pro14 := [0, 0, 0, 0, 0, 0, -245\,\mu_4^{13}, -3185\,\mu_4^{12}, -96\,\mu_5^{7}, -672\,\mu_5^{6}, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]$$

65

1

$Ac_{2,4}$

```
> pro15:=[0,0,0,0,0,0,-42*mu[4]^11,-462*mu[4]^10,28*mu[5]^6,168*mu[5]^5,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0];
> nops(pro15);
> ccA[2,4]:=convert(pro15,Vector):
> ccA[2,4][50];
```

$$pro15 := [0, 0, 0, 0, 0, 0, -42\,\mu_4^{11}, -462\,\mu_4^{10}, 28\,\mu_5^{6}, 168\,\mu_5^{5}, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]$$

65

1

$Ac_{3,4}$

```
> pro16:=[0,0,0,0,0,0,-7*mu[4]^9,-63*mu[4]^8,-8*mu[5]^5,-40*mu[5]^4,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0];
> nops(pro16);
> ccA[3,4]:=convert(pro16,Vector):
> ccA[3,4][51];
```

$$pro16 := [0, 0, 0, 0, 0, 0, -7\,\mu_4^{9}, -63\,\mu_4^{8}, -8\,\mu_5^{5}, -40\,\mu_5^{4}, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]$$

65

1

$Ac_{4,4}$

```
> pro17:=[1/mu[1]^4,-4/mu[1]^5,0,0,0,0,-mu[4]^7,-7*mu[4]^6,2*mu[5]^4,8*mu[5]^3,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,
0,0,0,0,0];
> nops(pro17);
> ccA[4,4]:=convert(pro17,Vector):
> ccA[4,4][52];
```

$$pro17 := [\mu_1^{-4}, -4\,\mu_1^{-5}, 0, 0, 0, 0, -\mu_4^{7}, -7\,\mu_4^{6}, 2\,\mu_5^{4}, 8\,\mu_5^{3}, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]$$

65

1

$Ac_{5,4}$

```
> pro18:=[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0];
> nops(pro18);
> ccA[5,4]:=convert(pro18,Vector):
> ccA[5,4][53];
```

$$pro18 := [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]$$

65

1

$Ac_{0,5}$

```
> pro19:=[0,0,0,0,0,0,11704*mu[4]^18,210672*mu[4]^17,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0];
> nops(pro19);
> ccA[0,5]:=convert(pro19,Vector):
> ccA[0,5][54];
```

$$pro19 := [0,0,0,0,0,0,11704\,\mu_4^{18},210672\,\mu_4^{17},0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0]$$

65

1

$Ac_{1,5}$

```
> pro20:=[0,0,0,0,0,0,1938*mu[4]^16,31008*mu[4]^15,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0];
> nops(pro20);
> ccA[1,5]:=convert(pro20,Vector):
> ccA[1,5][55];
```

$$pro20 := [0,0,0,0,0,0,1938\,\mu_4^{16},31008\,\mu_4^{15},0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0]$$

65

1

$Ac_{2,5}$

```
> pro21:=[0,0,0,0,0,0,320*mu[4]^14,4480*mu[4]^13,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0];
> nops(pro21);
> ccA[2,5]:=convert(pro21,Vector):
> ccA[2,5][56];
```

$$pro21 := [0,0,0,0,0,0,320\,\mu_4^{14},4480\,\mu_4^{13},0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0]$$

65

1

$Ac_{3,5}$

```
> pro22:=[0,0,0,0,0,0,52*mu[4]^12,624*mu[4]^11,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0];
> nops(pro22);
> ccA[3,5]:=convert(pro22,Vector):
> ccA[3,5][57];
```

$$pro22 := [0,0,0,0,0,0,52\,\mu_4^{12},624\,\mu_4^{11},0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0]$$

65

1

$Ac_{4,5}$

```
> pro23:=[0,0,0,0,0,0,8*mu[4]^10,80*mu[4]^9,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0];
> nops(pro23);
> ccA[4,5]:=convert(pro23,Vector):
> ccA[4,5][58];
```

$$pro23 := [0,0,0,0,0,0,8\,\mu_4^{10},80\,\mu_4^{9},0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0]$$

65

1

$Ac_{5,5}$

```
> pro24:=[1/mu[1]^5,-5/mu[1]^6,0,0,0,0,mu[4]^8,8*mu[4]^7,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0];
> nops(pro24);
> ccA[5,5]:=convert(pro24,Vector):
> ccA[5,5][59];
```

$$pro24 := [\mu_1^{-5}, -5\,\mu_1^{-6}, 0, 0, 0, 0, \mu_4^8, 8\,\mu_4^7, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0]$$

65

1

$Ac_{6,5}$

```
> pro25:=[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0];
> nops(pro25);
> ccA[6,5]:=convert(pro25,Vector):
> ccA[6,5][60];
```

$$pro25 := [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0]$$

65

1

Columns of $\mathscr{N}_c''$ (*twenty five* (10×1)-*vectors*)

```
> Mg:=(y,z,p,q)->(y,z+t*y^p/z^q);
```

$$Mg := (y, z, p, q) \mapsto y, z + \frac{t y^p}{z^q}$$

```
> M:=(y,z)->Mg(y,z,p,q);
```

$$M := (y, z) \mapsto Mg(y, z, p, q)$$

$Ac_{p,q}$

```
> cccA:=unapply(MatrixMatrixMultiply(R_2,champ2(M)),(p,q)):
```

Glueing

We form the 25 vectors of type c of $V(\mathfrak{D}_2)$.

```
> for q from 1 to 5 do
> for p from 0 to q+1 do
> for j from 1 to 10 do
> uu[p,q][j]:=cA(p,q)[j]
> od
> od
> od:
> for q from 1 to 5 do
> for p from 0 to q+1 do
> for j from 11 to 75 do
> uu[p,q][j]:=ccA[p,q][j-10]
> od
> od
> od:
> for q from 1 to 5 do
> for p from 0 to q+1 do
> for j from 76 to 85 do
> uu[p,q][j]:=cccA(p,q)[j-75]
> od
> od
> od:
> for q from 1 to 5 do
> for p from 0 to q+1 do
> uu[p,q]:=convert(uu[p,q],list)
> od
> od:
> q:=1:
> for p from 0 to q+1 do
> VectDef_c[p+1]:=uu[p,q]
> od:
```

```
> q:=2:
> for p from 0 to q+1 do
> VectDef_c[p+4]:=uu[p,q]
> od:
> q:=3:
> for p from 0 to q+1 do
> VectDef_c[p+8]:=uu[p,q]
> od:
> q:=4:
> for p from 0 to q+1 do
> VectDef_c[p+13]:=uu[p,q]
> od:
> q:=5:
> for p from 0 to q+1 do
> VectDef_c[p+19]:=uu[p,q]
> od:
> p:='p':
> q:='q':
```

1.6.4 Vectors of type d (we compute the columns of the matrices $\mathcal{N}'_{\mathrm{d}}$, $\mathcal{N}_{\mathrm{d}}$ and $\mathcal{N}''_{\mathrm{d}}$)

Columns of $\mathcal{N}'_{\mathrm{d}}$ *(five* (10×1)*-vectors)*

```
> Mg:=(y,z,p)->(y+t*y^(p+2)/z^p,z+t*y^(p+1)/z^(p-1));
```

$$Mg := (y,z,p) \mapsto y + \frac{ty^{p+2}}{z^p}, z + \frac{ty^{p+1}}{z^{p-1}}$$

```
> M:=(y,z)->Mg(y,z,p);
```

$$M := (y,z) \mapsto Mg(y,z,p)$$

Ad_p

```
> dA:=unapply(MatrixMatrixMultiply(R_1,champ1(M)),p):
```

Columns of $\mathcal{N}_{\mathrm{d}}$ *to which we add* 50 *zeros then a* (5×5)*-identity block*

Ad_1

```
> pro1:=[0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0];
> nops(pro1);
> ddA[1]:=convert(pro1,Vector):
> ddA[1][61];
```

$$pro1 := [0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0]$$

$$65$$

$$1$$

Ad_2

```
> pro2:=[0,0,mu[2]^2,2*mu[2],0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0];
> nops(pro2);
> ddA[2]:=convert(pro2,Vector):
> ddA[2][62];
```

$$pro2 := [0,0,\mu_2^2,2\,\mu_2,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0]$$

$$65$$

$$1$$

Ad_3

```
> pro3:=[0,0,0,0,mu[3]^2,2*mu[3],2*mu[4],2,2*mu[5],2,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0];
> nops(pro3);
> ddA[3]:=convert(pro3,Vector):
> ddA[3][63];
```

$pro3 := [0,0,0,0,\mu_3^2,2\,\mu_3,2\,\mu_4,2,2\,\mu_5,2,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,$
$0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0]$

65

1

Ad$_4$

```
> pro4:=[0,0,0,0,0,0,mu[4]^2,2*mu[4],0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0];
> nops(pro4);
> ddA[4]:=convert(pro4,Vector):
> ddA[4][64];
```

$pro4 := [0,0,0,0,0,0,\mu_4^2,2\,\mu_4,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,$
$0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0]$

65

1

Ad$_5$

```
> pro5:=[0,0,0,0,0,0,0,0,mu[5]^2,2*mu[5],0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1];
> nops(pro5);
> ddA[5]:=convert(pro5,Vector):
> ddA[5][65];
```

$pro5 := [0,0,0,0,0,0,0,0,\mu_5^2,2\,\mu_5,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,$
$0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1]$

65

1

Columns of $\mathscr{N}_{\mathrm{d}}''$ *(five* (10×1)*-vectors)*

```
> Mg:=(y,z,p)->(y+t*y^(p+2)/z^p,z+t*y^(p+1)/z^(p-1));
```

$$Mg := (y,z,p) \mapsto y+\frac{t y^{p+2}}{z^p}, z+\frac{t y^{p+1}}{z^{p-1}}$$

```
> M:=(y,z)->Mg(y,z,p);
```

$$M := (y,z) \mapsto Mg(y,z,p)$$

Ad$_p$

```
> dddA:=unapply(MatrixMatrixMultiply(R_2,champ2(M)),p):
```

Glueing

We form the 5 vectors of type d of $V(\mathfrak{D}_2)$.

```
> for k from 1 to 5 do
> for j from 1 to 10 do
> VectDef_d[k][j]:=dA(k)[j]
> od
> od:
> for k from 1 to 5 do
> for j from 11 to 75 do
> VectDef_d[k][j]:=ddA[k][j-10]
> od
> od:
> for k from 1 to 5 do
> for j from 76 to 85 do
> VectDef_d[k][j]:=dddA(k)[j-75]
> od
> od:
> for k from 1 to 5 do
> VectDef_d[k]:=convert(VectDef_d[k],list)
> od:
> for k from 1 to 5 do
> VectDef_d[k]:=convert(VectDef_d[k],Vector)
> od:
```

1.6.5 Construction of the matrix $\mathscr{V}_2$

We put together the 8 vectors of type a, the 25 vectors of type b, the 25 vectors of type c and the 5 vectors of type d.

```
> for k from 1 to 8 do
> V[k]:=VectDef_a[k]
> od:
> for k from 9 to 33 do
> V[k]:=VectDef_b[k-8]
> od:
> for k from 34 to 58 do
> V[k]:=VectDef_c[k-33]
> od:
> for k from 59 to 63 do
> V[k]:=VectDef_d[k-58]
> od:
> for k from 1 to 63 do
> WW[k]:=convert(V[k],list)
> od:
> MatVectDef:=Transpose(Matrix([seq(WW[k],k=1..63)])):
```

We define the images of the 8 vectors of $V(\mathfrak{D}_1)$ under the action of $\mathscr{Y}$.

```
> for k from 1 to 8 do
> ImageVectDef[k]:=MatrixMatrixMultiply(MM,VectDef[k])
> od:
```

We know how to compute these 8 vectors in the basis with the 65 vectors of $V(\mathfrak{D}_2)$ thanks to the relations (5.3) given in the article. We check that these relations are satisfied (these 8 instructions give 85 zeros).

```
> for k from 1 to 85 do
> simplify(ImageVectDef[1][k]-WW[1][k]-3*WW[62][k]+4*WW[15][k]+3*WW[36][k]);
> od:
> for k from 1 to 85 do
> simplify(ImageVectDef[2][k]-WW[2][k]+3*WW[60][k]);
> od:
> for k from 1 to 85 do
> simplify(ImageVectDef[3][k]-WW[3][k]+3*WW[59][k]);
> od:
> for k from 1 to 85 do
> simplify(ImageVectDef[4][k]-WW[4][k]-2*WW[20][k]-WW[40][k]+2*WW[63][k]);
> od:
> for k from 1 to 85 do
> simplify(ImageVectDef[5][k]-WW[5][k]-2*WW[61][k]);
> od:
> for k from 1 to 85 do
> simplify(ImageVectDef[6][k]-WW[6][k]-2*WW[60][k]);
> od:
> for k from 1 to 85 do
> simplify(ImageVectDef[7][k]-WW[7][k]);
> od:
```

1.7 Computation of the action of ψ_* on $\mathrm{H}^1(X, \mathrm{T}X)$

Construction of the supplementary $\mathscr{E}_1$ of $V(\mathfrak{D}_1)$ of dimension 22 in $W(\mathfrak{D}_1)$.

We form a (30×30)-matrix $\mathtt{NN}_1$ whose the first 22 columns are now zero and the last 8 are the basis vectors of $V(\mathfrak{D}_1)$ computed previously.

```
> for k from 1 to 30 do
> for j from 23 to 30 do
> N1[k][j]:=VectDef[j-22][k]
> od
> od:
> for k from 1 to 30 do
> for j from 1 to 22 do
> N1[k][j]:=0
> od
> od:
> for k from 1 to 30 do
> N1[k]:=convert(N1[k],list)
> od:
> NN1:=Matrix([seq(N1[k],k=1..30)]):
```

We fill the first 22 columns of $\mathtt{NN}_1$ by some vectors chosen among the algebraic basis of $W(\mathfrak{D}_1)$; they will define a supplementary $\mathscr{E}_1$ of $V(\mathfrak{D}_1)$.

```
> NN1[1,1]:=1:
> NN1[2,2]:=1:
> NN1[3,3]:=1:
> NN1[4,4]:=1:
> NN1[5,5]:=1:
> NN1[6,6]:=1:
> NN1[7,7]:=1:
> NN1[8,8]:=1:
> NN1[9,9]:=1:
> NN1[10,10]:=1:
> NN1[11,11]:=1:
> NN1[12,12]:=1:
> NN1[13,13]:=1:
> NN1[14,14]:=1:
> NN1[15,15]:=1:
> NN1[16,16]:=1:
> NN1[17,17]:=1:
> NN1[21,18]:=1:
> NN1[22,19]:=1:
> NN1[23,20]:=1:
> NN1[24,21]:=1:
> NN1[25,22]:=1:
```

We compute the determinant of $\mathtt{NN}_1$, that is nonzero for generic parameters μ_i; this shows that $\mathscr{E}_1$ is a supplementary to $V(\mathfrak{D}_1)$.

```
> Determinant(NN1);
```

$$12\,\mu_4\mu_5(-6\,\mu_4\mu_5 - 4\,\mu_4\mu_5^2 + 9\,\mu_5\alpha + 9\,\mu_4^2\alpha + 45\,\mu_5 + 45\,\mu_4^2 - 36\,\mu_4)$$

We extract the first 22 columns of $\mathtt{NN}_1$ in order to form the matrix `left`, that is the injection matrix of $\mathscr{E}_1$ into $W(\mathfrak{D}_1)$.

```
> for k from 1 to 30 do
> for j from 1 to 22 do
> coeff1[k][j]:=NN1[k,j]:
> od:
> od:
> for k from 1 to 30 do
> coeff1[k]:=convert(coeff1[k],list)
> od:
> left:=Matrix([seq(coeff1[k],k=1..30)]):
```

Definition of the supplementary $\mathscr{E}_2$ of $V(\mathfrak{D}_2)$ in $W(\mathfrak{D}_2)$ of dimension 22 which is the image of $\mathscr{E}_1$ by the inclusion of $W(\mathfrak{D}_1)$ into $W(\mathfrak{D}_2)$.

We form a (85×85)-matrix $\mathtt{NN}_2$ whose the first 22 columns are now zero and the last 63 are some basis vectors of $V(\mathfrak{D}_2)$ computed previously.

```
> for k from 1 to 85 do
> for j from 23 to 85 do
> N2[k][j]:=WW[j-22][k]
> od
> od:
> for k from 1 to 85 do
> for j from 1 to 22 do
> N2[k][j]:=0
> od
> od:
> for k from 1 to 85 do
> N2[k]:=convert(N2[k],list)
> od:
> NN2:=Matrix([seq(N2[k],k=1..85)]):
```

We fill the first 22 columns of $\mathtt{NN}_2$ by the basis vectors of $\mathscr{E}_2$.

```
> NN2[1,1]:=1:
> NN2[2,2]:=1:
> NN2[3,3]:=1:
> NN2[4,4]:=1:
> NN2[5,5]:=1:
```

```
> NN2[6,6]:=1:
> NN2[7,7]:=1:
> NN2[8,8]:=1:
> NN2[9,9]:=1:
> NN2[10,10]:=1:
> NN2[11,11]:=1:
> NN2[12,12]:=1:
> NN2[13,13]:=1:
> NN2[14,14]:=1:
> NN2[15,15]:=1:
> NN2[16,16]:=1:
> NN2[17,17]:=1:
> NN2[76,18]:=1:
> NN2[77,19]:=1:
> NN2[78,20]:=1:
> NN2[79,21]:=1:
> NN2[80,22]:=1:
```

The matrix `right` is the projection matrix of $W(\mathfrak{D}_2)$ onto $\mathscr{E}_2$.

```
> NNN2:=MatrixInverse(NN2):
> for k from 1 to 22 do
> for j from 1 to 85 do
> coeff2[k][j]:=NNN2[k,j]:
> od:
> od:
> for k from 1 to 22 do
> coeff2[k]:=convert(coeff2[k],list)
> od:
> right:=Matrix([seq(coeff2[k],k=1..22)]):
```

The matrix `OMG` is the matrix of f_* acting on $\mathrm{H}^1(X, \mathrm{T}X)$.

```
> OMG:=MatrixMatrixMultiply(right,MatrixMatrixMultiply(MM,left)):
> OMG;
```

$$\begin{bmatrix} \text{‘ 22 x 22 ‘ } (\textit{Matrix}) \\ \text{‘Data Type: ‘ } \textit{anything} \\ \text{‘Storage: ‘ } \textit{rectangular} \\ \text{‘Order: ‘ } \textit{Fortran\_order} \end{bmatrix}$$

We assign the parameters λ_i et μ_i otherwise the computation is too heavy.

```
> lambda[1]:=10:
> lambda[2]:=-6:
> lambda[3]:=5:
> lambda[4]:=2:
> lambda[5]:=3:
> mu[1]:=3:
> mu[2]:=-2:
> mu[3]:=4:
> mu[4]:=3:
> mu[5]:=29:
```

Computation of the eigenvalues of `OMG`.

```
> factor(CharacteristicPolynomial(OMG,x));
```

$$(x^2+3\,x+1)(x^2+18\,x+1)(x^2-7\,x+1)(x^2+x+1)(x-1)^2(x+1)^4(x^2-x+1)^4$$

We look at the size of the Jordan blocks for the multiple eigenvalues.

```
> Id:=Matrix(22,22,shape=identity):

> Rank(OMG+Id);
> Rank((OMG+Id)^2);
> Rank((OMG+Id)^3);
```

19

18

18

```
> Rank(OMG-Id);
> Rank((OMG-Id)^2);
```

20

20

```
> Rank(OMG-(1+I*sqrt(3))/2*Id);
> Rank((OMG-(1+I*sqrt(3))/2*Id^2));
```

19

19

2 Quadratic maps fixing a cuspidal cubic curve

```
> restart;
```

2.1 First definitions

2.1.1 The quadratic transform

Definition of the linear transform mapping $[1,1,1]$, $[a,1,a^3]$ and $[b,1,b^3]$ on $[1,0,0]$, $[0,1,0]$ and $[0,0,1]$.

```
> A:=(b^3-a^3)*x+(b*a^3-a*b^3)*y+(a-b)*z;
> B:=(1-b^3)*x+(b^3-b)*y+(b-1)*z;
> C:=(a^3-1)*x+(a-a^3)*y+(1-a)*z;
```

$$A := (b^3-a^3)x+(ba^3-ab^3)y+(a-b)z$$

$$B := (1-b^3)x+(b^3-b)y+(b-1)z$$

$$C := (a^3-1)x+(a-a^3)y+(1-a)z$$

Composition with the Cremona involution. The transformation $[x,y,z] \to [X,Y,Z]$ is a quadratic birational transformation whose indeterminacy points are $[1,1,1]$, $[a,1,a^3]$ and $[b,1,b^3]$.

```
> X:=B*C:
> Y:=A*C:
> Z:=A*B:
```

Definition of a 3x3 matrix.

```
> m[1]:=3*(a-b)^4*(b+1+4*a)^2*(2*b-1+2*a)^3*(2*a-b+2)^2*(b-1)*(b^2+2*b-b*a+2*a+4+a^2):
> m[2]:=-(3*(a-2-2*b))*(a-b)^2*(b+1+4*a)^2*(2*b-1+2*a)^2*(2*a-b+2)^2*(b-1)^3*(b^2+2*b-b*a+
2*a+4+a^2):
> m[3]:=3*(-1+a)^2*(b^2+2*b-b*a+2*a+4+a^2)*(b-1)*(2*a-b+2)^3*(2*b-1+2*a)^2*(a-b)^2*(b+1+4*
a)^2:
> m[4]:=-(2*a-b+2)^3*(2*b-1+2*a)^3*(b+1+4*a)^3*(a-b)^5:
> m[5]:=-(2*a-b+2)^3*(b+1+4*a)^3*(a-b)^3*(2*b-1+2*a)^3*(b-1)^2:
> m[6]:=-(2*b-1+2*a)^3*(a-b)^3*(b+1+4*a)^3*(2*a-b+2)^3*(-1+a)^2:
> m[7]:=27*(b^2+2*b-b*a+2*a+4+a^2)^3*(b-1)^3*(2*b-1+2*a)^3*(a-b)^2:
> m[8]:=-27*(b-1)^5*(b^2+2*b-b*a+2*a+4+a^2)^3*(a-2-2*b)^3:
> m[9]:=27*(b-1)^3*(2*a-b+2)^3*(-1+a)^2*(b^2+2*b-b*a+2*a+4+a^2)^3:
```

Introduction of an auxiliary coefficient `c`.

```
> c:=(1/9)*(2*a-b+2)*(2*b-1+2*a)*(b+1+4*a)*(a-b)/((b-1)*(b^2+2*b-b*a+2*a+4+a^2));
```

$$c := 1/9\,\frac{(2\,a-b+2)(2\,b-1+2\,a)(b+1+4\,a)(a-b)}{(b-1)(b^2+2\,b-ba+2\,a+4+a^2)}$$

Composition of $[x,y,z] \to [X,Y,Z]$ with m to obtain a birational transform fixing the cubic $y^2z = x^3$. Next we multiply by the diagonal element diag $\{c\mu, 1, (c\mu)^3\}$. The constant c has been chosen in order that μ is the multiplier of the transformation acting on the curve.

```
> U:=(c*mu)*(m[1]*X+m[2]*Y+m[3]*Z):
> V:=m[4]*X+m[5]*Y+m[6]*Z:
> W:=(c*mu)^3*(m[7]*X+m[8]*Y+m[9]*Z):
```

We express the previous transform in the affine coordinates (x,z).

```
> f:=unapply(U/V, x, z):
> y:=1:
> g:=unapply(W/V, x, z):
```

We check that the cubic $y^2z = x^3$ is fixed.

```
> simplify(g(t, t^3)-f(t, t^3)^3);
```

$$0$$

Computation of the multiplier.

```
> factor(f(t, t^3));
```

$$1/3\,(1+3\,t+b+a)\mu$$

Definition of the translation factor.

```
> epsilon:=(a+b+1)*mu/3;
```

$$\epsilon := 1/3\,(a+b+1)\mu$$

Definition of the points p_i^-.

```
> p1:=(1-2*a-2*b)*mu/3;
> p2:=(-2+a-2*b)*mu/3;
> p3:=(-2-2*a+b)*mu/3;
```

$$p1 := 1/3\,(-2\,b+1-2\,a)\mu$$

$$p2 := 1/3\,(a-2-2\,b)\mu$$

$$p3 := 1/3\,(b-2-2\,a)\mu$$

2.1.2 The fixed point

Computation of the smooth fixed point on the cubic.

```
> solve(f(t, t^3)-t, t);
```

$$-1/3\,\frac{(a+b+1)\mu}{\mu-1}$$

This point is (p, q) in the coordinates (x, z).

```
> p:=-(1/3)*mu*(a+b+1)/(-1+mu);
> q:=p^3;
```

$$p := -1/3\,\frac{(a+b+1)\mu}{\mu-1}$$

$$q := -1/27\,\frac{(a+b+1)^3\mu^3}{(\mu-1)^3}$$

We check that μ is an eigenvalue of the differential at a fixed point.

```
> factor((coeftayl(f(x,z), [x, z]=[p, q], [1, 0])-mu)*(coeftayl(g(x,z), [x, z]=[p, q], [0,
1])-mu)-coeftayl(g(x,z), [x, z]=[p, q], [1, 0])*coeftayl(f(x,z), [x, z]=[p, q], [0, 1]));
```

$$0$$

Calculation of the other eigenvalue: we divide the determinant by μ.

```
> zeta:=simplify(((coeftayl(f(x,z), [x, z]=[p, q], [1, 0]))*(coeftayl(g(x,z), [x, z]=[p, q],
[0, 1]))-coeftayl(g(x,z), [x, z]=[p, q], [1, 0])*coeftayl(f(x,z), [x, z]=[p, q], [0, 1])))
/mu:
```

2.2 Computations for the permutation (1)(2)(3)

```
> a:='a':  b:='b':
```

2.1.1 Computations of `a` and `b`

```
> solve({mu^(i-1)*(p1-p)+p=1, mu^(j-1)*(p2-p)+p=a}, {a, b});
```

$$\{a = \frac{2\,\mu^{j-1}\mu^2\mu^{i-1} - 3\,\mu^{j-1}\mu\,\mu^{i-1} - 2\,\mu^{j-1}\mu + \mu^{i-1}\mu - 1 + 3\,\mu^{j-1}}{(2\,\mu^{i-1}\mu - 3\,\mu^{i-1} + 1)\qquad(\mu^{j-1}\mu - 1)}, b = -\frac{\mu^{j-1}\mu^3\mu^{i-1} - 3\,\mu^{j-1}\mu^2\mu^{i-1} + 2\,\mu^{j-1}\mu^2 + 2\,\mu^{i-1}\mu^2}{(2\,\mu^{i-1}\mu - 3\,\mu^{i-1} + 1)\qquad\mu\,(\mu^{j-1}\mu - 1}$$

```
> a:=simplify((2*mu^(j-1)*mu^2*mu^(i-1)-3*mu^(j-1)*mu*mu^(i-1)-2*mu^(j-1)*mu+mu^(i-1)*mu-1+
3*mu^(j-1))/((2*mu^(i-1)*mu-3*mu^(i-1)+1)*(mu^(j-1)*mu-1)));
```

$$a := \frac{2\,\mu^{j+1+i} - 3\,\mu^{j+i} - 2\,\mu^{j+1} + \mu^{i+1} - \mu + 3\,\mu^j}{(\mu^j - 1)(2\,\mu^{i+1} - 3\,\mu^i + \mu)}$$

```
> b:=simplify(-(mu^(j-1)*mu^3*mu^(i-1)-3*mu^(j-1)*mu^2*mu^(i-1)+2*mu^(j-1)*mu^2+2*mu^(i-1)
*mu^2-5*mu+3)/((2*mu^(i-1)*mu-3*mu^(i-1)+1)*mu*(mu^(j-1)*mu-1)));
```

$$b := \frac{-\mu^{j+2+i} + 3\,\mu^{j+1+i} - 2\,\mu^{j+2} - 2\,\mu^{2+i} + 5\,\mu^2 - 3\,\mu}{\mu\,(\mu^j - 1)(2\,\mu^{i+1} - 3\,\mu^i + \mu)}$$

Verification: $\mu^{k-1} \times (p3 - p) + p = b$ when μ is a root of P_τ.

```
> factor(numer(expand((mu^(k-1)*(p3-p)+p-b))));
```

$$-3\,(\mu - 1)(2\,\mu - \mu^i\mu - \mu^j\mu - \mu^k\mu + \mu^k\mu\,\mu^j\mu^i + \mu^k\mu^i - 1 + \mu^j\mu^i - 2\,\mu^k\mu^j\mu^i + \mu^k\mu^j)$$

2.2.2 Computation of ζ

```
> factor(expand(zeta*mu^(i+j+k-3)-1));
```

$$-\frac{-\mu^i\mu - \mu^j\mu + 2\,\mu - \mu^k\mu + \mu^k\mu\,\mu^j\mu^i + \mu^k\mu^i + \mu^k\mu^j + \mu^j\mu^i - 2\,\mu^k\mu^j\mu^i - 1}{-\mu^i\mu + 2\,\mu - \mu^j\mu - 1 + \mu^j\mu^i}$$

2.3 Computations for the permutation (123)

```
> a:='a':  b:='b':
```

2.3.1 Computations of a and b

```
> solve({mu^(i-1)*(p1-p)+p=a, mu^(j-1)*(p2-p)+p=b}, {a, b});
```

$$\{a = \frac{(2\,\mu^{j-1}\mu^2\mu^{i-1} + 2\,\mu^{i-1}\mu - 3\,\mu^{j-1}\mu\,\mu^{i-1} - 1)\mu}{2\,\mu^{j-1}\mu^3\mu^{i-1} - 3\,\mu^{j-1}\mu^2\mu^{i-1} + 2\,\mu^{i-1}\mu^2 + 3\,\mu^{j-1}\mu^2 - 3\,\mu^{i-1}\mu - 3\,\mu^{j-1}\mu + 5\,\mu - 3}, b = -\frac{(\mu^{j-1}\mu^2\mu^{i-1} - 3\,\mu^{j-1}\mu\,\mu^{i-1} + \mu^{i-1}\mu - 3\,\mu^{j-1} + 1 + 3\,\mu^{j-1}\mu)\mu}{2\,\mu^{j-1}\mu^3\mu^{i-1} - 3\,\mu^{j-1}\mu^2\mu^{i-1} + 2\,\mu^{i-1}\mu^2 + 3\,\mu^{j-1}\mu^2 - 3\,\mu^{i-1}\mu - 3\,\mu^{j-1}\mu + 5\,\mu - 3}\}$$

```
> a:=(2*mu^(j-1)*mu^2*mu^(i-1)+2*mu^(i-1)*mu-3*mu^(j-1)*mu*mu^(i-1)-1)*mu/(2*mu^(j-1)*mu^3*
mu^(i-1)-3*mu^(j-1)*mu^2*mu^(i-1)+2*mu^(i-1)*mu^2+3*mu^(j-1)*mu^2-3*mu^(i-1)*mu-3*mu^(j-1)
*mu+5*mu-3);
```

$$a := \frac{(2\,\mu^{j-1}\mu^2\mu^{i-1} + 2\,\mu^{i-1}\mu - 3\,\mu^{j-1}\mu\,\mu^{i-1} - 1)\mu}{2\,\mu^{j-1}\mu^3\mu^{i-1} - 3\,\mu^{j-1}\mu^2\mu^{i-1} + 2\,\mu^{i-1}\mu^2 + 3\,\mu^{j-1}\mu^2 - 3\,\mu^{i-1}\mu - 3\,\mu^{j-1}\mu + 5\,\mu - 3}$$

```
> b:=-(mu^(j-1)*mu^2*mu^(i-1)-3*mu^(j-1)*mu*mu^(i-1)+mu^(i-1)*mu-3*mu^(j-1)+1+3*mu^(j-1)*
mu)*mu/(2*mu^(j-1)*mu^3*mu^(i-1)-3*mu^(j-1)*mu^2*mu^(i-1)+2*mu^(i-1)*mu^2+3*mu^(j-1)*mu^2-3*
mu^(i-1)*mu-3*mu^(j-1)*mu+5*mu-3);
```

$$b := -\frac{(\mu^{j-1}\mu^2\mu^{i-1} - 3\,\mu^{j-1}\mu\,\mu^{i-1} + \mu^{i-1}\mu - 3\,\mu^{j-1} + 1 + 3\,\mu^{j-1}\mu)\mu}{2\,\mu^{j-1}\mu^3\mu^{i-1} - 3\,\mu^{j-1}\mu^2\mu^{i-1} + 2\,\mu^{i-1}\mu^2 + 3\,\mu^{j-1}\mu^2 - 3\,\mu^{i-1}\mu - 3\,\mu^{j-1}\mu + 5\,\mu - 3}$$

Verification: $\mu^{k-1} \times (p3 - p) + p = 1$ when μ is a root of P_τ.

```
> factor(numer(expand((mu^(k-1)*(p3-p)+p-1))));
```

$$-3\,(\mu-1)(\mu^k\mu\,\mu^i+\mu^k\mu\,\mu^j+\mu^j\mu\,\mu^i+\mu^i\mu+2\,\mu+\mu^j\mu+\mu^k\mu+\mu^k\mu\,\mu^j\mu^i-\mu^k\mu^i-\mu^i-\mu^k\mu^j-\mu^k-\mu^j\mu^i-1-\mu^j-2\,\mu^k\mu^j\mu^i)$$

2.3.2 Computations of ζ

```
> factor(expand(zeta*mu^(i+j+k-3)-1));
```

$$-\frac{\mu^i\mu+2\,\mu+\mu^k\mu+\mu^j\mu\,\mu^i+\mu^j\mu+\mu^k\mu\,\mu^j\mu^i+\mu^k\mu\,\mu^j+\mu^k\mu\,\mu^i-\mu^k\mu^j-\mu^i-1-\mu^j-\mu^k-\mu^k\mu^i-2\,\mu^k\mu^j\mu^i-\mu^j\mu^i}{\mu^i\mu+2\,\mu+\mu^j\mu\,\mu^i+\mu^j\mu-\mu^i-1-\mu^j-\mu^j\mu^i}$$

2.4 Computations for the permutation (1)(23)

```
> a:='a':  b:='b':
```

2.4.1 Computations of `a` and `b`

```
> solve({mu^(i-1)*(p1-p)+p=1, mu^(j-1)*(p2-p)+p=b}, {a, b});
```

$$\{a = \frac{2\,\mu^{j-1}\mu^3\mu^{i-1}-2\,\mu^{j-1}\mu^2+2\,\mu^{i-1}\mu^2-3\,\mu^{j-1}\mu^2\mu^{i-1}-5\,\mu+3\,\mu^{j-1}\mu+3}{(2\,\mu^{i-1}\mu-3\,\mu^{i-1}+1)(\mu^{j-1}\mu+1)\mu}, b = -\frac{\mu^{j-1}\mu^2\mu^{i-1}+2\,\mu^{j-1}\mu+\mu^{i-1}\mu-3\,\mu^{j-1}\mu\,\mu^{i-1}-1}{(2\,\mu^{i-1}\mu-3\,\mu^{i-1}+1)(\mu^{j-1}\mu+1)}\}$$

```
> a:=(2*mu^(j-1)*mu^3*mu^(i-1)-3*mu^(j-1)*mu^2*mu^(i-1)+2*mu^(i-1)*mu^2-2*mu^(j-1)*mu^2-5*
mu+3*mu^(j-1)*mu+3)/((-3*mu^(i-1)+2*mu^(i-1)*mu+1)*(mu^(j-1)*mu+1)*mu);
```

$$a := \frac{2\,\mu^{j-1}\mu^3\mu^{i-1}-2\,\mu^{j-1}\mu^2+2\,\mu^{i-1}\mu^2-3\,\mu^{j-1}\mu^2\mu^{i-1}-5\,\mu+3\,\mu^{j-1}\mu+3}{(2\,\mu^{i-1}\mu-3\,\mu^{i-1}+1)\qquad(\mu^{j-1}\mu+1)\mu}$$

```
> b:=-(mu^(j-1)*mu^2*mu^(i-1)-3*mu^(j-1)*mu*mu^(i-1)+mu^(i-1)*mu+2*mu^(j-1)*mu-1)/((-3*mu^(
i-1)+2*mu^(i-1)*mu+1)*(mu^(j-1)*mu+1));
```

$$b := -\frac{\mu^{j-1}\mu^2\mu^{i-1}+2\,\mu^{j-1}\mu+\mu^{i-1}\mu-3\,\mu^{j-1}\mu\,\mu^{i-1}-1}{(2\,\mu^{i-1}\mu-3\,\mu^{i-1}+1)\qquad(\mu^{j-1}\mu+1)}$$

Verification: $\mu^{k-1}\times(p_3-p)+p=a$ when μ is a root of P_τ.

```
> factor(numer(expand((mu^(k-1)*(p3-p)+p-a))));
```

$$-3\,\mu\,(\mu-1)(-2\,\mu+\mu^j\mu\,\mu^i+\mu^i\mu-\mu^j\mu-\mu^k\mu+\mu^k\mu\,\mu^i+\mu^k\mu\,\mu^j\mu^i+\mu^j+1-\mu^j\mu^i+\mu^k\mu^j-\mu^k\mu^i-2\,\mu^k\mu^j\mu^i+\mu^k)$$

2.4.2 Computations of ζ

```
> factor(expand(zeta*mu^(i+j+k-3)-1));
```

$$-\frac{\mu^i\mu-2\,\mu-\mu^k\mu+\mu^j\mu\,\mu^i-\mu^j\mu+\mu^k\mu\,\mu^j\mu^i+\mu^k\mu\,\mu^i+\mu^j+\mu^k\mu^j+1+\mu^k-\mu^k\mu^i-2\,\mu^k\mu^j\mu^i-\mu^j\mu^i}{\mu^j\mu\,\mu^i+\mu^i\mu-\mu^j\mu^i-2\,\mu-\mu^j\mu+\mu^j+1}$$