Step by step computation of NC Grobner basis for ideal annihilating the Arveson boundary of a free quadrilateral

NOTE: This notebook requires the NCAlgebra package available at http://math.ucsd.edu/~ncalg/.
Relevant paper: https://arxiv.org/abs/2008.13250

Preliminaries

We perform the computation using NCAlgebra. The key step in computing a NC Grobner basis is computing and reducing the so called basic S-polynomials obtained from a collection of NC polynomials.

To simplify our computation, we set w:=c0+c1 z1+c2 z2 and adjoin 1-w**(c0+c1 z1+c2 z2) and 1-(c0+c1 z1+c2 z2)**w to the relations defining our ideal. We will work in the pure lexiographic monomial order z1 << z2 << w.

A brief description of the main functions add follows.

RuleToPoly[rule] converts a rule for replacement to a NC polynomial. E.g. RuleToPoly[w -> z1**z2]= w-z1**z2.

NCRRR[list,rus,iters] is shorthand for “NCReplaceRepeated Repeated”. This command iteratively applies NCReplaceRepeated followed by NCExpand to a collection of polynomials. This prevents NCReplaceRepeated from missing replaceable phrases which are “hidden” by distribution. E.g. the monomials z1**z2**z1 and z1**z1**z2 are not recognized by NCReplaceRepeated in an expression of the form z1**(z2**z1+z1**z2)

spoly[l1,ind1,r1,l2,ind2,r2,polys] is used for computing S-polynomials between given members in the list “polys”. Computation and reduction of S-polynomials forms the major step of each GB iteration. On each iteration it is sufficient to consider only “basic” S-polynomials, i.e. S-polynoimials which either have l2=r1=1 or l1=r2=1. An S-polynomial is between poly1 and poly2 is formed by multiplying poly1 and poly2 by minimal monomials so that the leading monomials of the products are equal.

In[∘]:=
```
<< NC`
<< NCAlgebra`
<< NCGBX`
RuleToPoly[rule_] := NCReplaceRepeated[rule, Rule → Subtract];
view[list_] := NCReplaceRepeated[list, NonCommutativeMultiply → Dot] // MatrixForm
NCRRR[list_, rus_, iters_] := Block[{currlist, i},
  currlist = list;
  For[i = 1, i ≤ iters, i++,
   currlist = NCE[Simplify[NCRR[currlist, rus]]]];
  Return[currlist]]
spoly[l1_, ind1_, r1_, l2_, ind2_, r2_, polys_] :=
 NCE[l1 ** polys[[ind1]] ** r1 - l2 ** polys[[ind2]] ** r2]

SNC[z1, z2]
SetMonomialOrder[z1, z2, w]
Clear[c0, c1, c2]
spoly[l1_, ind1_, r1_, l2_, ind2_, r2_, polys_] :=
 NCE[l1 ** polys[[ind1]] ** r1 - l2 ** polys[[ind2]] ** r2]
```

```
You are using the version of NCAlgebra which is found in:

  C:\Users\Eric\NC\

You can now use "<< NCAlgebra`" to load NCAlgebra.

------------------------------------------------------------------
NCAlgebra - Version 5.0.4
Compatible with Mathematica Version 10 and above

Authors:

  J. William Helton*
  Mauricio de Oliveira&

* Math, UCSD, La Jolla, CA
& MAE, UCSD, La Jolla, CA

with major earlier contributions by:

  Mark Stankus$
  Robert L. Miller#

$ Math, Cal Poly San Luis Obispo
# General Atomics Corp

Copyright:
  Helton and de Oliveira 2017
  Helton 2002
  Helton and Miller June 1991
  All rights reserved.

The program was written by the authors and by:
  David Hurst, Daniel Lamm, Orlando Merino, Robert Obar,
  Henry Pfister, Mike Walker, John Wavrik, Lois Yu,
  J. Camino, J. Griffin, J. Ovall, T. Shaheen, John Shopple.
  The beginnings of the program come from eran@slac.
  Considerable recent help came from Igor Klep.

Current primary support is from the
  NSF Division of Mathematical Sciences.

This program was written with support from
  AFOSR, NSF, ONR, Lab for Math and Statistics at UCSD,
  UCSD Faculty Mentor Program,
  and US Department of Education.

For NCAlgebra updates see:

  www.github.com/NCAlgebra/NC
  www.math.ucsd.edu/~ncalg

------------------------------------------------------------------
```

Out[*]= {{z1}, {z2}, {w}}

To simplify our computation, we set w:=c0+c1 z1+c2 z2 and adjoin 1-w**(c0+c1 z1+c2 z2) and 1-(c0+c1 z1+c2 z2)**w to the relations defining our ideal.

In[∘]:=
```
InvPolys = {c0 + c1 * z1 + c2 * z2 - z1 ** w ** z1, c0 + c1 * z1 + c2 * z2 - z2 ** w ** z2,
    1/c2 - c0 w/c2 - c1 z1 ** w/c2 - z2 ** w, 1/c2 - c0 w/c2 - c1 w ** z1/c2 - w ** z2};
InvRus = {z1 ** w ** z1 → c0 + c1 * z1 + c2 * z2, z2 ** w ** z2 → c0 + c1 * z1 + c2 * z2,
    z2 ** w → 1/c2 - c0 w/c2 - c1 z1 ** w/c2, w ** z2 → 1/c2 - c0 w/c2 - c1 w ** z1/c2};
LeadMons = {z1 ** w ** z1, z2 ** w ** z2, z2 ** w, w ** z2};
ColumnForm[InvPolys]
ColumnForm[LeadMons]
```

Out[∘]=

$$c0 + c1\,z1 + c2\,z2 - z1 \ast\ast w \ast\ast z1$$
$$c0 + c1\,z1 + c2\,z2 - z2 \ast\ast w \ast\ast z2$$
$$\frac{1}{c2} - \frac{c0\,w}{c2} - \frac{c1\,z1\ast\ast w}{c2} - z2 \ast\ast w$$
$$\frac{1}{c2} - \frac{c0\,w}{c2} - \frac{c1\,w\ast\ast z1}{c2} - w \ast\ast z2$$

Out[∘]=

$$z1 \ast\ast w \ast\ast z1$$
$$z2 \ast\ast w \ast\ast z2$$
$$z2 \ast\ast w$$
$$w \ast\ast z2$$

We now form all basic S-polynomials for the appearing in InvPolys, and we reduce the resulting polynomials by InvRus.

In[∘]:=
```
spoly1 = spoly[LeadMons[[2]], 1, 1, 1, 2, LeadMons[[1]], InvPolys];
spoly2 = spoly[1, 1, LeadMons[[2]], LeadMons[[1]], 2, 1, InvPolys];
spoly3 = spoly[1, 1, LeadMons[[3]], LeadMons[[1]], 3, 1, InvPolys];
spoly4 = spoly[LeadMons[[3]], 1, 1, 1 , 3, LeadMons[[1]], InvPolys];
spoly5 = spoly[1, 1, LeadMons[[4]], LeadMons[[1]], 4, 1, InvPolys];
spoly6 = spoly[LeadMons[[4]], 1, 1, 1 , 4, LeadMons[[1]], InvPolys];
spoly7 = spoly[1, 2, 1, 1, 3, z2, InvPolys];
spoly8 = spoly[1, 2, w, z2 ** w, 3, 1, InvPolys];
spoly9 = spoly[1, 2, 1, z2, 4, 1, InvPolys];
spoly10 = spoly[w, 2, 1, 1, 4, w ** z2, InvPolys];
spoly11 = spoly[w, 3, 1, 1, 4, w, InvPolys];
spoly12 = spoly[1, 3, z2, z2, 4, 1, InvPolys];
RedSpolys =
  DeleteDuplicates[DeleteCases[NCRRR[{spoly1, spoly2, spoly3, spoly4, spoly5, spoly6,
      spoly7, spoly8, spoly9, spoly10, spoly11, spoly12}, InvRus, 10], 0]];
Length[RedSpolys];
RedSpolys // ColumnForm
```

Out[∘]=

$$\frac{c0}{c2} - \frac{c0\,c1^2}{c2} + c0\,c2 - \frac{c0^2\,w}{c2} + \frac{c1\,z1}{c2} - \frac{c1^3\,z1}{c2} + c1\,c2\,z1 - z2 - c1^2\,z2 + c2^2\,z2 - \frac{c0\,c1\,w\ast\ast z1}{c2} - \frac{c0\,c1\,z1\ast\ast w}{c2}$$
$$c0 + \frac{c0}{c2^2} - \frac{c0\,c1^2}{c2^2} - \frac{c0^2\,w}{c2^2} + c1\,z1 + \frac{c1\,z1}{c2^2} - \frac{c1^3\,z1}{c2^2} - \frac{z2}{c2} - \frac{c1^2\,z2}{c2} + c2\,z2 - \frac{c0\,c1\,w\ast\ast z1}{c2^2} - \frac{c0\,c1\,z1\ast\ast w}{c2^2}$$
$$1 - \frac{1}{c2^2} - \frac{c1^2}{c2^2} + \frac{2\,c0\,w}{c2^2} - \frac{c0^2\,w\ast\ast w}{c2^2} + \frac{2\,c1\,z1\ast\ast w}{c2^2} - \frac{c0\,c1\,w\ast\ast z1\ast\ast w}{c2^2} - \frac{c0\,c1\,z1\ast\ast w\ast\ast w}{c2^2}$$
$$1 - \frac{1}{c2^2} - \frac{c1^2}{c2^2} + \frac{2\,c0\,w}{c2^2} - \frac{c0^2\,w\ast\ast w}{c2^2} + \frac{2\,c1\,w\ast\ast z1}{c2^2} - \frac{c0\,c1\,w\ast\ast w\ast\ast z1}{c2^2} - \frac{c0\,c1\,w\ast\ast z1\ast\ast w}{c2^2}$$

The first and second reduced S-polynomials are constant multiples of each other. Additionally we see that the lead monomial in tfirst polynoimal w**z1, while the lead monomial in the third and fourth polynomials is w**z1**w, hence the third and fourth polynomials are reduced by the first. We adjoin the rule corresponding to the first polynomial to InvRus (while performing a sanity check to verify the rule is correct), and reduce the remaining S-polynomials. All remaining S-polynomials now reduce to zero.

In[∘]:=
```
RedSpolys[[1]] / (-c0 * c1) * (c2) // Simplify // Expand
RedSpolys[[2]] / (-c0 * c1) * (c2^2) // Simplify // Expand
```

Out[∘]= $-\frac{1}{c1} + c1 - \frac{c2^2}{c1} + \frac{c0\,w}{c1} - \frac{z1}{c0} + \frac{c1^2\,z1}{c0} - \frac{c2^2\,z1}{c0} + \frac{c2\,z2}{c0\,c1} + \frac{c1\,c2\,z2}{c0} - \frac{c2^3\,z2}{c0\,c1} + w ** z1 + z1 ** w$

Out[∘]= $-\frac{1}{c1} + c1 - \frac{c2^2}{c1} + \frac{c0\,w}{c1} - \frac{z1}{c0} + \frac{c1^2\,z1}{c0} - \frac{c2^2\,z1}{c0} + \frac{c2\,z2}{c0\,c1} + \frac{c1\,c2\,z2}{c0} - \frac{c2^3\,z2}{c0\,c1} + w ** z1 + z1 ** w$

In[∘]:= $-\left(-\frac{1}{c1} + c1 - \frac{c2^2}{c1} + \frac{c0\,w}{c1} - \frac{z1}{c0} + \frac{c1^2\,z1}{c0} - \frac{c2^2\,z1}{c0} + \frac{c2\,z2}{c0\,c1} + \frac{c1\,c2\,z2}{c0} - \frac{c2^3\,z2}{c0\,c1} + z1 ** w\right)$

Out[∘]= $\frac{1}{c1} - c1 + \frac{c2^2}{c1} - \frac{c0\,w}{c1} + \frac{z1}{c0} - \frac{c1^2\,z1}{c0} + \frac{c2^2\,z1}{c0} - \frac{c2\,z2}{c0\,c1} - \frac{c1\,c2\,z2}{c0} + \frac{c2^3\,z2}{c0\,c1} - z1 ** w$

In[∘]:= GBIter1Join1 = Append[InvRus,

$w ** z1 \to \frac{1}{c1} - c1 + \frac{c2^2}{c1} - \frac{c0\,w}{c1} + \frac{z1}{c0} - \frac{c1^2\,z1}{c0} + \frac{c2^2\,z1}{c0} - \frac{c2\,z2}{c0\,c1} - \frac{c1\,c2\,z2}{c0} + \frac{c2^3\,z2}{c0\,c1} - z1 ** w$];

```
NCRR[RedSpolys[[1]], GBIter1Join1[[5]]] // Simplify
```

Out[∘]= 0

In[∘]:=
```
NCRRR[RedSpolys, GBIter1Join1, 3]
```

Out[∘]= {0, 0, 0, 0}

In[∘]:=
```
GBIter1Join1 // view
```

Out[∘]//MatrixForm=

$$\begin{pmatrix} z1.w.z1 \to c0 + c1\,z1 + c2\,z2 \\ z2.w.z2 \to c0 + c1\,z1 + c2\,z2 \\ z2.w \to \frac{1}{c2} - \frac{c0\,w}{c2} - \frac{c1\,z1.w}{c2} \\ w.z2 \to \frac{1}{c2} - \frac{c0\,w}{c2} - \frac{c1\,w.z1}{c2} \\ w.z1 \to \frac{1}{c1} - c1 + \frac{c2^2}{c1} - \frac{c0\,w}{c1} + \frac{z1}{c0} - \frac{c1^2\,z1}{c0} + \frac{c2^2\,z1}{c0} - \frac{c2\,z2}{c0\,c1} - \frac{c1\,c2\,z2}{c0} + \frac{c2^3\,z2}{c0\,c1} - z1.w \end{pmatrix}$$

We now reduce the our initial relations by our new relation.
The Relation z1.w.z1→c0+c1 z1+c2 z2 is reduced by
$w.z1 \to \frac{1}{c1} - c1 + \frac{c2^2}{c1} - \frac{c0\,w}{c1} + \frac{z1}{c0} - \frac{c1^2\,z1}{c0} + \frac{c2^2\,z1}{c0} - \frac{c2\,z2}{c0\,c1} - \frac{c1\,c2\,z2}{c0} + \frac{c2^3\,z2}{c0\,c1} - z1.w$.
The relation z2.w.z2→c0+c1 z1+c2 z2 is reduced by both $w.z2 \to \frac{1}{c2} - \frac{c0\,w}{c2} - \frac{c1\,w.z1}{c2}$ and $z2.w \to \frac{1}{c2} - \frac{c0\,w}{c2} - \frac{c1\,z1.w}{c2}$. We reduce by $w.z2 \to \frac{1}{c2} - \frac{c0\,w}{c2} - \frac{c1\,w.z1}{c2}$ since it is higher in the monomial order. The result is reduced by
$w.z1 \to \frac{1}{c1} - c1 + \frac{c2^2}{c1} - \frac{c0\,w}{c1} + \frac{z1}{c0} - \frac{c1^2\,z1}{c0} + \frac{c2^2\,z1}{c0} - \frac{c2\,z2}{c0\,c1} - \frac{c1\,c2\,z2}{c0} + \frac{c2^3\,z2}{c0\,c1} - z1.w$.

In[∘]:=
```
NCRRR[RuleToPoly[GBIter1Join1[[1]]], GBIter1Join1[[5]], 5]
NCRRR[NCRRR[RuleToPoly[GBIter1Join1[[2]]], GBIter1Join1[[4]], 5],
   GBIter1Join1[[5]], 5] * c2 / c1 // Expand
```

Out[∘]=
$$-c0+\frac{z1}{c1}-2\,c1\,z1+\frac{c2^2\,z1}{c1}-c2\,z2-\frac{c0\,z1**w}{c1}+\frac{z1**z1}{c0}-\frac{c1^2\,z1**z1}{c0}+\frac{c2^2\,z1**z1}{c0}-\frac{c2\,z1**z2}{c0\,c1}-\frac{c1\,c2\,z1**z2}{c0}+\frac{c2^3\,z1**z2}{c0\,c1}-z1**z1**w$$

Out[∘]=
$$-\frac{c0\,c2}{c1}-c2\,z1+c1\,z2-\frac{2\,c2^2\,z2}{c1}-\frac{z2**z1}{c0}+\frac{c1^2\,z2**z1}{c0}-\frac{c2^2\,z2**z1}{c0}+\frac{c2\,z2**z2}{c0\,c1}+\frac{c1\,c2\,z2**z2}{c0}-\frac{c2^3\,z2**z2}{c0\,c1}+z2**z1**w$$

In[∘]:=
```
GBIter1Reduct =
  Join[{z1 ** z1 ** w → -c0 + z1/c1 - 2 c1 z1 + c2^2 z1/c1 - c2 z2 - c0 z1 ** w/c1 + z1 ** z1/c0 - c1^2 z1 ** z1/c0 +
      c2^2 z1 ** z1/c0 - c2 z1 ** z2/(c0 c1) - c1 c2 z1 ** z2/c0 + c2^3 z1 ** z2/(c0 c1),
    z2 ** z1 ** w → c0 c2/c1 + c2 z1 - c1 z2 + 2 c2^2 z2/c1 + z2 ** z1/c0 - c1^2 z2 ** z1/c0 +
      c2^2 z2 ** z1/c0 - c2 z2 ** z2/(c0 c1) - c1 c2 z2 ** z2/c0 + c2^3 z2 ** z2/(c0 c1)}, GBIter1Join1[[3 ;; 5]]];
```

A sanity check is performed to verify the new rules have been formed correctly.

In[∘]:=
```
NCRR[NCRRR[RuleToPoly[GBIter1Join1[[1]]], GBIter1Join1[[5]], 5],
  GBIter1Reduct[[1]]] // Simplify
NCRR[NCRRR[NCRRR[RuleToPoly[GBIter1Join1[[2]]], GBIter1Join1[[4]], 5],
    GBIter1Join1[[5]], 5] * c2 / c1 // Expand, GBIter1Reduct[[2]]] // Simplify
```

Out[∘]= 0

Out[∘]= 0

This completes the first major GB iteration and reduction.

GB Iteration 2

We now preform our second GB iteration.

In[∘]:=
```
GBp1 = RuleToPoly[GBIter1Reduct];
% // view
```

Out[∘]//MatrixForm=
$$\left(\begin{array}{c}
c0-\frac{z1}{c1}+2\,c1\,z1-\frac{c2^2\,z1}{c1}+c2\,z2+\frac{c0\,z1.w}{c1}-\frac{z1.z1}{c0}+\frac{c1^2\,z1.z1}{c0}-\frac{c2^2\,z1.z1}{c0}+\frac{c2\,z1.z2}{c0\,c1}+\frac{c1\,c2\,z1.z2}{c0}-\frac{c2^3\,z1.z2}{c0\,c1}+z1. \\
-\frac{c0\,c2}{c1}-c2\,z1+c1\,z2-\frac{2\,c2^2\,z2}{c1}-\frac{z2.z1}{c0}+\frac{c1^2\,z2.z1}{c0}-\frac{c2^2\,z2.z1}{c0}+\frac{c2\,z2.z2}{c0\,c1}+\frac{c1\,c2\,z2.z2}{c0}-\frac{c2^3\,z2.z2}{c0\,c1}+z2.z1.w \\
-\frac{1}{c2}+\frac{c0\,w}{c2}+\frac{c1\,z1.w}{c2}+z2.w \\
-\frac{1}{c2}+\frac{c0\,w}{c2}+\frac{c1\,w.z1}{c2}+w.z2 \\
-\frac{1}{c1}+c1-\frac{c2^2}{c1}+\frac{c0\,w}{c1}-\frac{z1}{c0}+\frac{c1^2\,z1}{c0}-\frac{c2^2\,z1}{c0}+\frac{c2\,z2}{c0\,c1}+\frac{c1\,c2\,z2}{c0}-\frac{c2^3\,z2}{c0\,c1}+w.z1+z1.w
\end{array}\right)$$

In[•]:=
```
lm = {z1 ** z1 ** w, z2 ** z1 ** w, z2 ** w, w ** z2, w ** z1};
ColumnForm[lm]
```

Out[•]=
```
z1 ** z1 ** w
z2 ** z1 ** w
z2 ** w
w ** z2
w ** z1
```

We again form S-polynomials from our relations. In total four new relations are generated. Note: The 3rd and 4th polynomials have not changed, so there is no need to consider an S-polynomial between them.

In[•]:=
```
sp12lr = spoly[lm[[2]], 1, 1, 1, 2, lm[[1]], GBp1];
sp12rl = spoly[1, 1, lm[[2]], lm[[1]], 2, 1, GBp1];
sp13lr = spoly[lm[3], 1, 1, 1, 3, lm[1], GBp1];
sp13rl = spoly[1, 1, lm[3], lm[1], 3, 1, GBp1];
sp14lr = spoly[lm[4], 1, 1, 1, 4, lm[1], GBp1];
sp14rl = spoly[1, 1, z2, z1 ** z1, 4, 1, GBp1];
sp15lr = spoly[w, 1, 1, 1, 5, z1 ** w, GBp1];
sp15rl = spoly[1, 1, z1, z1 ** z1 , 5, 1, GBp1];
sp23lr = spoly[lm[3], 2, 1, 1, 3, lm[2], GBp1];
sp23rl = spoly[1, 2, lm[3], lm[2], 3, 1, GBp1];
sp24lr = spoly[w, 2, 1, 1, 4, z1 ** w, GBp1];
sp24rl = spoly[1, 2, z2, z2 ** z1, 4, 1, GBp1];
sp25lr = spoly[lm[[5]], 2, 1, 1, 5, lm[[2]], GBp1];
sp25rl = spoly[1, 2, z1, z2 ** z1 , 5, 1, GBp1];
sp35lr = spoly[lm[[5]], 3, 1, 1, 5, lm[[3]], GBp1];
sp35rl = spoly[1, 3, z1, z2, 5, 1, GBp1];
sp45lr = spoly[lm[[5]], 4, 1, 1, 5, lm[4], GBp1];
sp45rl = spoly[1, 4, lm[[5]], lm[[4]], 5, 1, GBp1];

RedSpolys =
  DeleteDuplicates[DeleteCases[NCRRR[{sp12lr, sp12rl, sp13lr, sp13rl, sp14lr, sp14rl,
      sp15lr, sp15rl, sp23lr, sp23rl, sp24lr, sp24rl, sp25lr, sp25rl,
      sp35lr, sp35rl, sp45lr, sp45rl}, GBIter1Reduct, 10], 0]];
Length[RedSpolys]
RedSpolys // ColumnForm
```

Out[•]= 4

Out[•]=

$$\frac{c0^2}{c2}+\frac{3\,c0\,c1\,z1}{c2}-\frac{c0\,c2\,z1}{c1}+2\,c0\,z2-\frac{z1{**}z1}{c2}+\frac{3\,c1^2\,z1{**}z1}{c2}-2\,c2\,z1{**}z1+3\,c1\,z1{**}z2-\frac{2\,c2^2\,z1{**}z2}{c1}+c1\,z2{**}$$
$$\frac{c0^2}{c1}+3\,c0\,z1+\frac{c0\,c2\,z2}{c1}-\frac{z1{**}z1}{c1}+3\,c1\,z1{**}z1-\frac{c2^2\,z1{**}z1}{c1}+c2\,z1{**}z2+c2\,z2{**}z1-\frac{z1{**}z1{**}z1}{c0}+\frac{c1^2\,z1{**}z1}{c0}$$
$$-\frac{c0^2}{c1}-2\,c0\,z1+\frac{c0\,c1\,z2}{c2}-\frac{3\,c0\,c2\,z2}{c1}-c1\,z1{**}z1-c2\,z1{**}z2+\frac{2\,c1^2\,z2{**}z1}{c2}-3\,c2\,z2{**}z1+\frac{z2{**}z2}{c1}+2\,c1\,z2$$
$$-\frac{c0\,c2\,z1}{c1}+c0\,z2-c2\,z1{**}z1+2\,c1\,z2{**}z1-\frac{2\,c2^2\,z2{**}z1}{c1}+c2\,z2{**}z2-\frac{z2{**}z1{**}z1}{c0}+\frac{c1^2\,z2{**}z1{**}z1}{c0}-\frac{c2^2\,z2{*}}{c}$$

The first polynomial in this list is reduced by the second, as the lead monomial of the second polynomial is z1**z2**z1, and this monomial appears in the third polynomial. Neither the first nor the third polynomial reduces the others. Similarly, the third polynomial is reduced by the fourth. No other polynomial is reduced by another polynomial. We first make rules corresponding to the second and fourth polynomial, then reduce the second polynomial. A sanity checked is performed to ensure the rules correctly correspond to the appropriate polynomials. We then reduce the remaining polynomials.

In[•]:=
```
GBIter2Join1 = {z1 ** z2 ** z1 → - 1/(-(1 + c1^2) c2 + c2^3)
    (c0 (1 - 3 c1^2 + c2^2) z1 ** z1 - c0 (c0 (c0 + 3 c1 z1 + c2 z2) + c1 c2 (z1 ** z2 + z2 ** z1)) +
     c1 (1 - c1^2 + c2^2) z1 ** z1 ** z1), z2 ** z2 ** z1 → - 1/(-(1 + c1^2) c2 + c2^3)
    (c0 (c0 c2 z1 - c0 c1 z2 + c1 c2 z1 ** z1 + 2 (-c1^2 + c2^2) z2 ** z1 - c1 c2 z2 ** z2) +
     c1 (1 - c1^2 + c2^2) z2 ** z1 ** z1)};
NCRR[RedSpolys[[2]], GBIter2Join1[[1]]] // Simplify
NCRR[RedSpolys[[4]], GBIter2Join1[[2]]] // Simplify
RedSpolys2reduct = NCRRR[{RedSpolys[[1]], RedSpolys[[3]]}, GBIter2Join1, 5]
```

Out[•]= 0

Out[•]= 0

Out[•]=
$$\left\{-\frac{c0\,c2\,z1}{c1}+c0\,z2-c2\,z1**z1+2\,c1\,z1**z2-\frac{2\,c2^2\,z1**z2}{c1}+c2\,z2**z2-\frac{z1**z1**z2}{c0}+\frac{c1^2\,z1**z1**z2}{c0}-\frac{c2^2\,z1**z1**z2}{c0}+\frac{c2\,z1**z2**z2}{c0\,c1}+\frac{c1\,c2\,z1**z2**z2}{c0}-\frac{c2^3\,z1**z2**z2}{c0\,c1},\right.$$
$$\left.-\frac{c0^2}{c1}-c0\,z1-\frac{3\,c0\,c2\,z2}{c1}-c2\,z1**z2-c2\,z2**z1+\frac{z2**z2}{c1}+c1\,z2**z2-\frac{3\,c2^2\,z2**z2}{c1}-\frac{z2**z1**z2}{c0}+\frac{c1^2\,z2**z1**z2}{c0}-\frac{c2^2\,z2**z1**z2}{c0}+\frac{c2\,z2**z2**z2}{c0\,c1}+\frac{c1\,c2\,z2**z2**z2}{c0}-\frac{c2^3\,z2**z2**z2}{c0\,c1}\right\}$$

The resulting polynomial do not reduce each other and cannot be reduced by any member of GBIter2Join1 or any member of GBIter1Reduct. Also, these new rules cannot reduce members of GBIter1Reduct, since all degree three terms in those polynomials contain the variable w. We now create rules for the reduced polynomials and perform a sanity check to ensure the rule has been made correctly.

In[•]:=
```
GBIter2Join2 = {z1 ** z2 ** z2 → - 1/(-(1 + c1^2) c2 + c2^3)
    (c0 (c0 c2 z1 - c0 c1 z2 + c1 c2 z1 ** z1 + 2 (-c1^2 + c2^2) z1 ** z2 - c1 c2 z2 ** z2) +
     c1 (1 - c1^2 + c2^2) z1 ** z1 ** z2), z2 ** z2 ** z2 →
   - 1/(-(1 + c1^2) c2 + c2^3) (c0^2 (c0 + c1 z1 + 3 c2 z2) + c0 c1 c2 (z1 ** z2 + z2 ** z1) -
     c0 (1 + c1^2 - 3 c2^2) z2 ** z2 + c1 (1 - c1^2 + c2^2) z2 ** z1 ** z2)};
NCRR[RedSpolys2reduct[[1]], GBIter2Join2[[1]]] // Simplify
NCRR[RedSpolys2reduct[[2]], GBIter2Join2[[2]]] // Simplify
```

Out[•]= 0

Out[•]= 0

Joining together all relations completes the second major GB iteration and reduction.

In[•]:=
```
GBIter2Reduct = Join[GBIter1Reduct, GBIter2Join1, GBIter2Join2];
```

GB Iteration 3

In fact, GBIter2Reduct is a NC Grobner basis. We verify this by showing that there are no remaining obstructions.

In[ ]:=
```
GBp2 = RuleToPoly[GBIter2Reduct];
% // view
```

Out[ ]//MatrixForm=

$$
\left(\begin{array}{c}
c0-\frac{z1}{c1}+2\,c1\,z1-\frac{c2^2\,z1}{c1}+c2\,z2+\frac{c0\,z1.w}{c1}-\frac{z1.z1}{c0}+\frac{c1^2\,z1.z1}{c0}-\frac{c2^2\,z1.z1}{c0}+\frac{c2\,z1.z2}{c0\,c1}+\frac{c1\,c2\,z1.z2}{c0}-\frac{c2^3\,z1.z2}{c0\,c1}+z1. \\
-\frac{c0\,c2}{c1}-c2\,z1+c1\,z2-\frac{2\,c2^2\,z2}{c1}-\frac{z2.z1}{c0}+\frac{c1^2\,z2.z1}{c0}-\frac{c2^2\,z2.z1}{c0}+\frac{c2\,z2.z2}{c0\,c1}+\frac{c1\,c2\,z2.z2}{c0}-\frac{c2^3\,z2.z2}{c0\,c1}+z2.z1.w \\
-\frac{1}{c2}+\frac{c0\,w}{c2}+\frac{c1\,z1.w}{c2}+z2.w \\
-\frac{1}{c2}+\frac{c0\,w}{c2}+\frac{c1\,w.z1}{c2}+w.z2 \\
-\frac{1}{c1}+c1-\frac{c2^2}{c1}+\frac{c0\,w}{c1}-\frac{z1}{c0}+\frac{c1^2\,z1}{c0}-\frac{c2^2\,z1}{c0}+\frac{c2\,z2}{c0\,c1}+\frac{c1\,c2\,z2}{c0}-\frac{c2^3\,z2}{c0\,c1}+w.z1+z1.w \\
\frac{c0\,(1-3\,c1^2+c2^2)\,z1.z1-c0\,(c0\,(c0+3\,c1\,z1+c2\,z2)+c1\,c2\,(z1.z2+z2.z1))+c1\,(1-c1^2+c2^2)\,z1.z1.z1}{(-1-c1^2)\,c2+c2^3}+z1.z2.z1 \\
\frac{c0\,(c0\,c2\,z1-c0\,c1\,z2+c1\,c2\,z1.z1+2\,(-c1^2+c2^2)\,z2.z1-c1\,c2\,z2.z2)+c1\,(1-c1^2+c2^2)\,z2.z1.z1}{(-1-c1^2)\,c2+c2^3}+z2.z2.z1 \\
\frac{c0\,(c0\,c2\,z1-c0\,c1\,z2+c1\,c2\,z1.z1+2\,(-c1^2+c2^2)\,z1.z2-c1\,c2\,z2.z2)+c1\,(1-c1^2+c2^2)\,z1.z1.z2}{(-1-c1^2)\,c2+c2^3}+z1.z2.z2 \\
\frac{c0^2\,(c0+c1\,z1+3\,c2\,z2)+c0\,c1\,c2\,(z1.z2+z2.z1)-c0\,(1+c1^2-3\,c2^2)\,z2.z2+c1\,(1-c1^2+c2^2)\,z2.z1.z2}{(-1-c1^2)\,c2+c2^3}+z2.z2.z2
\end{array}\right.
$$

In[ ]:=
```
lm = {z1 ** z1 ** w, z2 ** z1 ** w, z2 ** w, w ** z2,
   w ** z1, z1 ** z2 ** z1, z2 ** z2 ** z1, z1 ** z2 ** z2, z2 ** z2 ** z2};
ColumnForm[
 lm]
```

Out[ ]=
```
z1 ** z1 ** w
z2 ** z1 ** w
z2 ** w
w ** z2
w ** z1
z1 ** z2 ** z1
z2 ** z2 ** z1
z1 ** z2 ** z2
z2 ** z2 ** z2
```

Since the first 5 polynomials have not changed, there is no need to consider S-polynomials which are formed from a combination of these polynomials. We form the remaining S-polynomials and show the are reduced by GBIter2Reduct.

In[ ]:=
```
sp16lr = spoly[z1 ** z2, 1, 1, 1, 6, z1 ** w, GBp2];
sp16rl = spoly[1, 1, lm[[6]], lm[[1]], 6, 1, GBp2];
sp17lr = spoly[z2 ** z2, 1, 1, 1, 7, z1 ** w, GBp2];
sp17rl = spoly[1, 1, lm[[7]], lm[[1]], 7, 1, GBp2];
sp18lr = spoly[lm[[8]], 1, 1, 1, 8, lm[[1]], GBp2];
sp18rl = spoly[1, 1, lm[[8]], lm[[1]], 8, 1, GBp2];
sp19lr = spoly[lm[[9]], 1, 1, 1, 9, lm[[1]], GBp2];
sp19rl = spoly[1, 1, lm[[9]], lm[[1]], 9, 1, GBp2];
sp26lr = spoly[z1, 2, 1, 1, 6, w, GBp2];
sp26rl = spoly[1, 2, lm[[6]], lm[[2]], 6, 1, GBp2];
sp27lr = spoly[z2, 2, 1, 1, 7, w, GBp2];
sp27rl = spoly[1, 2, lm[[7]], lm[[2]], 7, 1, GBp2];
```

```
sp28lr = spoly[z1 ** z2, 2, 1, 1, 8, z1 ** w, GBp2];
sp28rl = spoly[1, 2, lm[[8]], lm[[2]], 8, 1, GBp2];
sp29lr = spoly[z2 ** z2, 2, 1, 1, 9, z1 ** w, GBp2];
sp29rl = spoly[1, 2, lm[[9]], lm[[2]], 9, 1, GBp2];
sp36lr = spoly[lm[[6]], 3, 1, 1, 6, lm[[3]], GBp2];
sp36rl = spoly[1, 3, lm[[6]], lm[[3]], 6, 1, GBp2];
sp37lr = spoly[lm[[7]], 3, 1, 1, 7, lm[[3]], GBp2];
sp37rl = spoly[1, 3, lm[[7]], lm[[3]], 7, 1, GBp2];
sp38lr = spoly[z1 ** z2, 3, 1, 1, 8, w, GBp2];
sp38rl = spoly[1, 3, lm[[8]], lm[[3]], 8, 1, GBp2];
sp39lr = spoly[z2 ** z2, 3, 1, 1, 9, w, GBp2];
sp39rl = spoly[1, 3, lm[[9]], lm[[3]], 9, 1, GBp2];
sp46lr = spoly[lm[[6]], 4, 1, 1, 6, lm[[4]], GBp2];
sp46rl = spoly[1, 4, lm[[6]], lm[[4]], 6, 1, GBp2];
sp47lr = spoly[lm[[7]], 4, 1, 1, 7, lm[[4]], GBp2];
sp47rl = spoly[1, 4, z2 ** z1, w, 7, 1, GBp2];
sp48lr = spoly[lm[[8]], 4, 1, 1, 8, lm[[4]], GBp2];
sp48rl = spoly[1, 4, lm[[8]], lm[[4]], 8, 1, GBp2];
sp49lr = spoly[lm[[9]], 4, 1, 1, 9, lm[[4]], GBp2];
sp49rl = spoly[1, 4, z2 ** z2, w, 9, 1, GBp2];
sp56lr = spoly[lm[[6]], 5, 1, 1, 6, lm[[5]], GBp2];
sp56rl = spoly[1, 5, z2 ** z1, w, 6, 1, GBp2];
sp57lr = spoly[lm[[7]], 5, 1, 1, 7, lm[[5]], GBp2];
sp57rl = spoly[1, 5, lm[[7]], lm[[5]], 7, 1, GBp2];
sp58lr = spoly[lm[[8]], 5, 1, 1, 8, lm[[5]], GBp2];
sp58rl = spoly[1, 5, z2 ** z2, w, 8, 1, GBp2];
sp59lr = spoly[lm[[9]], 5, 1, 1, 9, lm[[5]], GBp2];
sp59rl = spoly[1, 5, lm[[9]], lm[[5]], 9, 1, GBp2];
sp67lr = spoly[z2 ** z2, 6, 1, 1, 8, z2 ** z1, GBp2];
sp67rl = spoly[1, 6, lm[[7]], lm[[6]], 7, 1, GBp2];
sp68lr = spoly[lm[[8]], 6, 1, 1, 8, lm[[6]], GBp2];
sp68rl = spoly[1, 6, z2 ** z2, z1 ** z2, 8, 1, GBp2];
sp69lr = spoly[lm[[9]], 6, 1, 1, 9, lm[[6]], GBp2];
sp69rl = spoly[1, 6, lm[[9]], lm[[6]], 9, 1, GBp2];
sp78lr = spoly[z1, 7, 1, 1, 8, z1, GBp2];
sp78rl = spoly[1, 7, z2 ** z2, z2 ** z2, 8, 1, GBp2];
sp79lr = spoly[z2, 7, 1, 1, 9, z1, GBp2];
sp79rl = spoly[1, 7, lm[[9]], lm[[7]], 9, 1, GBp2];
sp89lr = spoly[lm[[9]], 8, 1, 1, 9, lm[[9]], GBp2];
sp89rl = spoly[1, 8, z2, z1, 9, 1, GBp2];


splist = {sp16lr, sp16rl, sp17lr, sp17rl, sp18lr, sp18rl, sp19lr, sp19rl, sp26lr, sp26rl,
   sp27lr, sp27rl, sp28lr, sp28rl, sp29lr, sp29rl, sp36lr, sp36rl, sp37lr, sp37rl, sp38lr,
   sp38rl, sp39lr, sp39rl, sp46lr, sp46rl, sp47lr, sp47rl, sp48lr, sp48rl, sp49lr, sp49rl,
   sp56lr, sp56rl, sp57lr, sp57rl, sp58lr, sp58rl, sp59lr, sp59rl, sp67lr, sp67rl,
   sp68lr, sp68rl, sp69lr, sp69rl, sp78lr, sp78rl, sp79lr, sp79rl, sp89lr, sp89rl};

Length[splist]

RedSpolys =
  DeleteDuplicates[DeleteCases[NCRRR[splist, GBIter2Reduct, 10], 0]] // FullSimplify;
Length[RedSpolys]
RedSpolys // ColumnForm
```

Out[ ]= 52

Out[ ]= 0

Out[ ]=

Since all basic S-polynoimals are reduced by GBIter2Reduct, we conclude that GBIter2Reduct is a NC Grobner basis for the initial relations. As in the the GB computed by NCGBX, we have four relations which are NC polynomials in z1 and z2.

```
In[ ]:= NCRRR[InvPolys, GBIter2Reduct, 3]
Out[ ]= {0, 0, 0, 0}
```

Comparison to results using NCGBX

We now compare to results obtained using NCGBX. First we use NCGBX as a second check that we indeed have a NC Grobner basis.

```
In[ ]:= GBcheck = NCMakeGB[GBIter2Reduct];

* * * * * * * * * * * * * * * *
* * *   NCPolyGroebner    * * *
* * * * * * * * * * * * * * * *
* Symbolic coefficients detected
* Monomial order: z1 ≪ z2 ≪ w
* Reduce and normalize initial set
> Initial set could not be reduced
* Computing initial set of obstructions
* Found Groebner basis with 9 polynomials
* * * * * * * * * * * * * * * *

In[ ]:= Simplify[RuleToPoly[GBcheck] - RuleToPoly[GBIter2Reduct]]
Out[ ]= {0, 0, 0, 0, 0, 0, 0, 0, 0}
```

Additionally we compute a NC Grobner basis from the initial relations using NCGBX and show that the resulting GB reduces GBIter2Reduct, and also that GBIter2Reduct reduces the resulting GB, hence these GBs generate the same ideal.

```
In[ ]:= GBtest = NCMakeGB[InvPolys];
```

```
* * * * * * * * * * * * * * * *
* * *   NCPolyGroebner    * * *
* * * * * * * * * * * * * * * *
* Symbolic coefficients detected
* Monomial order: z1 ≪ z2 ≪ w
* Reduce and normalize initial set
> Initial set could not be reduced
* Computing initial set of obstructions
> MAJOR Iteration 1, 7 polys in the basis, 11 obstructions
> MAJOR Iteration 2, 9 polys in the basis, 14 obstructions
* Found Groebner basis with 9 polynomials
* * * * * * * * * * * * * * * *
```

In[∘]:=
```
NCRRR[RuleToPoly[GBtest], GBIter2Reduct, 5]
NCRRR[RuleToPoly[GBIter2Reduct], GBtest, 5]
```

Out[∘]= {0, 0, 0, 0, 0, 0, 0, 0, 0}

Out[∘]= {0, 0, 0, 0, 0, 0, 0, 0, 0}