In[6]:=
```
(* First get the package and its dependencies. *)
<< NC`
<< NCAlgebra`
<< SDP`
<< NCSE`
<< NCSEBackwardsCompatible` (* only needed for files created before 22Aug2018 *)
```

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

C:\Users\ericm\NC\

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

```
(* This is a notebook designed to help understand the NCSE package. *)
```

```
(* This is a circle of radius 1, defined by a 2-tuple A1,
A2 of 2 by 2 matrices at level 1. The spectrahedron defined by this tuple at
   level 1 is formed from the linear matrix inequality I-A1*x1-A2*x2 ≥ 0*)
```

```
CircA = {{{0, 1}, {1, 0}}, {{1, 0}, {0, -1}}};
```

```
ViewTuple[CircA]
```

$\left\{\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}\right\}$

```
(* Let's find an extreme point on this circle at level 1. We just want
 to see the point and the linear functional optimized to find this point,
so let's set the diagnostic level, which controls how much information is output,
to 1 (the default is 4). *)
CircAExtremePoint = FindExtremePoint[CircA, 1, DiagnosticLevel → 1]
```

$\left\{\{\{\{0.226808\}\}, \{\{-0.97394\}\}\}, \left\{-\frac{17}{100}, \frac{73}{100}\right\}\right\}$

```
(* The first output list is the extreme point Xopt and the second output list
 is the coefficients of the linear functional used to generate that point *)
```

```
(* Since CircAExtremePoint[[1]] is supposed to be an extreme point on a circle
 of radius 1 at level 1, we'd expect the norm of the extreme point to be 1. *)
```

```
TupleNorm[CircAExtremePoint[[1]]]
```

1.

```
(* We would also expect that this extreme point is Arveson and Euclidean extreme. *)
ArvesonTest[CircA, CircAExtremePoint[[1]]]
```

$\{\text{True}, \{2.37545 \times 10^{-12}, 1.\}\}$

```
EuclideanTest[CircA, CircAExtremePoint[[1]]]
```

$\{\text{True}, \{2.37545 \times 10^{-12}, 1.\}\}$

```
(* The actual linear functional can be viewed by using MakeFunctionalRational. The 2 in
  the argument corresponds to the fact that CircA is a 2-tuple. The 1 in the argument
  corresponds to the fact that we're at level 1 of the free spectrahedron. *)
MakeFunctionalRational[2, 1, WeightVector → CircAExtremePoint[[2]]]
```

$$\left\{\left\{-\frac{17}{100}, \frac{73}{100}\right\}, -\frac{17}{100} X[1,1,1] + \frac{73}{100} X[2,1,1]\right\}$$

```
(* Suppose you want to generate this point again somehow using
 the same linear functional, just use the option WeightVector. Let's
 get more information this time by using the diagnostic level 2. *)

CircAExtremePointCopy =
 FindExtremePoint[CircA, 1, WeightVector → CircAExtremePoint[[2]], DiagnosticLevel → 2]
```

$$\left\{\{\{\{0.226808\}\}, \{\{-0.97394\}\}\}, \left\{-\frac{17}{100}, \frac{73}{100}\right\}, \text{True}, \text{True}\right\}$$

```
(* The output is the same as CircAExtremePoint,
except with two "True" values at the end. This is because the
 extreme point found is both an Arveson point and an Euclidean point. *)

(* Now let's randomly generate 10 points
 on the circle at level 1 with a new seed, say 23. *)

CircA10ExtremePoints = FindExtremeAndAnalyze[CircA, 1, 10, 23];

{{1, 10}}

10

{{{1, 1, 10}}}

False

0

False

{{}, 0}

0

0

(* There are 9 lines of output. According to 2nd line "10" of this output,
we have 10 irreducible Arveson points in total, and looking at the first line,
"{{1,10}}" all of them are at level 1 with L[XOpt] having kernel dimension 1.

 According to the second 3 lines of this output,
we have no irreducible Euclidean points that aren't Arveson. The 7th
 line "{{},0}" implies that there are no numerically bad points. The 8th
 line "0" implies that there are no reducible points. The 9th line "0"
 are the points that are not extreme (this will almost always be 0). *)

(* Let's look at level 2 now. *)

FindExtremeAndAnalyze[CircA, 2, 50, 23];
```

```
{}

0

False

False

0

False

{{}, 0}

50

0

(* There aren't any irreducible Euclidean extreme points;
looking at the 2nd to last line,
all 50 points are reducible. But if we're curious as to whether they are Arveson or not,
let's use the AnalyzeIrredOnly option. *)

FindExtremeAndAnalyze[CircA, 2, 50, 23, AnalyzeIrredOnly → False];

{{2, 50}}

0

{{}}

False

0

False

{{}, 0}

50

0

(* The first line of this output shows that all 50 points found were all Arveson extreme
 points and 2nd to last line shows that all 50 points found are reducible. *)

(* Let's study an irreducible bounded spectrahedron. We can use the
 MakeIrreducibleBoundedA to generate one. Let's use the seed 979012. *)
IrredA = MakeIrreducibleBoundedA[5, 4, 979012];
FindExtremeAndAnalyze[IrredA, 2, 20, 23];

{{3, 11}, {4, 1}}

12

{{{3, 8, 11}}, {{4, 4, 1}}}

False

0

False

{{}, 0}

8

0
```

```
(* From this output, we see that we found 12 irreducible Arveson points at level 2,
split up into 11 points with L[XOpt] having kernel dimension 3 and 1 point with
 L[XOpt] having kernel dimension 4. The third line of this output shows that
 all 11 Arveson extreme points with kernel dimension 3 have tangent dimension
 8 and the Arveson point with kernel dimension 4 have tangent dimension 4. *)
```

More Advanced Commands and Options

```
(*(?? using WeightMatrix ??)*)
```