▪ Lemma 4.3: Computation for Claim 2

In[1]:=
```
SetDirectory["~/writing/WIP/KappaLib/"];
<< KappaLib.m
```

```
KappaLib v1.1
```

▪ By Lemma 4.3 (i)-(ii), we can represent kappa, using a symmetric invertible 3x3 matrix A and an antisymmetric 3x3 matrix K.

In[3]:=
```
A = emGeneralSymmetric3x3["a"];
K = {
    { 0,  k1, k2},
   {-k1,  0, k3},
   {-k2, -k3, 0}
  };
```

In[5]:=
```
 (* define adjugate of A and verify.
   *
   * Note: For invertible A, adj A = (det A) A^(-1).
   *
   *)
adjA = Table[
            (-1) ^ (i + j) Det[Drop[A, {i, i}, {j, j}]],
            {j, 1, 3}, {i, 1, 3}
            ];

Union[Simplify[Flatten[adjA.A - Det[A] IdentityMatrix[3]]]]
Union[Simplify[Flatten[A.adjA - Det[A] IdentityMatrix[3]]]]
Union[Simplify[Flatten[K + Transpose[K]]]]
```

Out[6]= {0}

Out[7]= {0}

Out[8]= {0}

▪ Define A,B,C,D and (2,2)-tensor kappa induced by A and K in Lemma 4.2.

We know that det(A) != 0. To simplify expressions, let us introduce variable

alpha = det (A)

In[9]:=
```
invA = 1 / alpha adjA;
AA = A;
BB = - invA. (IdentityMatrix[3] + K.invA.K.invA);
CC = K.invA;
DD = -invA.K;

kappa = emABCDToKappa[AA, BB, CC, DD];
```

▪ Define the 4x4 matrix G0:

In[15]:=
```
kVec[i_] := Sum[
  A[[i]][[b]] 1 / 2 Signature[{b, c, d}] K[[c]][[d]],
  {b, 1, 3}, {c, 1, 3}, {d, 1, 3}
 ]

G0 = Table[0, {i, 1, 4}, {j, 1, 4}];

G0[[1]][[1]] = alpha;

For[k = 1, k ≤ 3, k++,
  G0[[1]][[k + 1]] = kVec[k];
  G0[[k + 1]][[1]] = kVec[k];
 ];

For[k = 1, k ≤ 3, k++,
  For[l = 1, l ≤ 3, l++,
    G0[[k + 1]][[l + 1]] = -A[[k]][[l]] + 1 / alpha kVec[k] kVec[l];
   ];
 ];
```

- **Check that Det[G0] = -alpha^2 (or hence Det[G] = -1).**
  As described in the proof, this can also be shown using a Shur complement.

In[20]:=
```
FullSimplify[(Det[G0] + alpha^2) /. alpha → Det[AA]]
```

Out[20]= 0

- **Claim 2:**

  kappa = - sgn (det(A)) ast_h (**)

  where h is the (0,2) tensor with entries given by G^-1.

- **In the paper the the RHS in (*) is expanded into**

  -sgn det(A) (ast_h)^ij_rs = -1/alpha G0^ia G0^jb epsilon_abrs

In[21]:=
```
RHS = emZeroKappa[];

    For[i = 1, i <= 4, i++,
        For[j = i + 1, j <= 4, j++,
            For[r = 1, r <= 4, r++,
                For[s = r + 1, s <= 4, s++,
                    RHS[[i]][[j]][[r]][[s]] = -1 / alpha Sum[

              G0[[i]][[a]] G0[[j]][[b]] Signature[{a, b, r, s}],
                                  {a, 1, 4}, {b, 1, 4}
                              ];
                ];
            ];
        ];
    ];
```

In[23]:=
```
(* Compute LHS-RHS in (**), and cancel all alpha:s in denominators *)
delta = Expand[(kappa - RHS) alpha^3];
delta = Union[Flatten[delta]];

(* Check that the result is a polynomial *)
Union[
 Table[
  PolynomialQ[delta[[i]], Variables[delta[[i]]]],
  {i, 1, Length[delta]}
 ]
]
```

Out[25]= {True}

In[26]:=
```
Simplify[delta /. alpha → Det[A]] // Timing
```

Out[26]= {0.023346, {0, 0, 0, 0, 0, 0, 0, 0, 0}}