These are the *Mathematica* solutions for the paper:

“Permutation-type solutions to the Yang-Baxter and other n-simplex equations” by J. Hietarinta.

computations by Dara O Shayda

dara@lossofgenerality.com

In[1]:=
```
<< Notation`
```

This formulation is incomprehensible and I guessed how it works:

In order to write the n-simplex equations in terms of A and B let us further define (in analogue with (2))

$$(A_{K_1 \ldots K_n})_i^j = \begin{cases} A_\alpha^\beta, & \text{if } i = K_\alpha,\ j = K_\beta \text{ for some } \alpha,\ \beta, \\ \delta_i^j, & \text{otherwise,} \end{cases} \tag{11}$$

$$(B_{K_1 \ldots K_n})_i = \begin{cases} B_i, & \text{if } i = K_\alpha, \text{ for some } \alpha, \\ 0, & \text{otherwise,} \end{cases} \tag{12}$$

so that

$$(R_{K_1 \ldots K_n})_{i_1 \ldots i_N}^{j_1 \ldots j_N} = \prod_{\mu=1}^{N} \delta^{j_n}_{(A_{K_1 \ldots K_n})_\mu^\nu i_\nu + (B_{K_1 \ldots K_n})_\mu}, \tag{13}$$

where now the ν summation runs from 1 to N.

```
In[2]:= Notation[ R_{i_,j_} ⟹ makeA[A, {i_, j_}, 3] ]
        Notation[ R_{i_,j_,k_} ⟹ makeA[A, {i_, j_, k_}, 6] ]
        Notation[ R_{i_,j_,k_,h_} ⟹ makeA[A, {i_, j_, k_, h_}, 10] ]

        makeMatrix[dim_, a_] := Array[Subscript[a, #1, #2] &, dim]

        (* makeA generates the A matrices in Hietarinta paper *)

        makeA[A_, list_, n_] := Module[{res, i, j, k},

          res = ConstantArray[-1, {n, n}];

          Table[If[MemberQ[list, i] != True, res [[i]][[j]] = KroneckerDelta[i, j];
            res [[j]][[i]] = KroneckerDelta[j, i] ], {i, 1, n}, {j, 1, n}];

          Table[k = 1; Table[
            If[res[[list[[i]]]][[j]] == -1, res[[list[[i]]]][[j]] = A[[i]][[k]]; k = k + 1];,
            {j, 1, n}], {i , 1, Length[list]}];
          res

         ]
```

2-Simplex

Make a 2x2 matrix:

```
In[7]:= A = makeMatrix[{2, 2}, a];
        A // MatrixForm
```

Out[8]//MatrixForm=

$$\begin{pmatrix} a_{1,1} & a_{1,2} \\ a_{2,1} & a_{2,2} \end{pmatrix}$$

Rename the entries to match the paper:

```
In[9]:=
        A = A /. {a_{1,1} → a, a_{1,2} → b, a_{1,3} → c, a_{2,1} → c, a_{2,2} → d};
        A // MatrixForm
```

Out[10]//MatrixForm=

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix}$$

Make the R matrices for 2-simplex:

In[11]:=

```
R_{1,2} // MatrixForm
```

Out[11]//MatrixForm=

$$\begin{pmatrix} a & b & 0 \\ c & d & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

In[12]:=

```
R_{1,3} // MatrixForm
```

Out[12]//MatrixForm=

$$\begin{pmatrix} a & 0 & b \\ 0 & 1 & 0 \\ c & 0 & d \end{pmatrix}$$

In[13]:=

```
R_{2,3} // MatrixForm
```

Out[13]//MatrixForm=

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & a & b \\ 0 & c & d \end{pmatrix}$$

The matrix equation:

In[14]:=

```
(R_{1,2}.R_{1,3}. R_{2,3} - R_{2,3}.R_{1,3}.R_{1,2}) // MatrixForm
```

Out[14]//MatrixForm=

$$\begin{pmatrix} 0 & a\,b\,c & -b + b^2 + a\,b\,d \\ -a\,b\,c & -b^2\,c + b\,c^2 & b\,c\,d \\ c - c^2 - a\,c\,d & -b\,c\,d & 0 \end{pmatrix}$$

Solve the matrix equation:

In[15]:=

```
sol = Solve[R_{1,2}.R_{1,3}. R_{2,3} - R_{2,3}.R_{1,3}.R_{1,2} == ConstantArray[0, {3, 3}], {a, b, c, d}];
```

In[16]:=

```
Table[ A /. sol[[i]] // MatrixForm , {i, 1, Length[sol]}]
```

Out[16]=

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