Computational Companion to "Quasi-Symmetric Nets: A Constructive Approach to the Equimodular Elliptic Type of Kokotsakis Polyhedra" — Criterion Helper

A. Nurmatov, M. Skopenkov, F. Rist, J. Klein, D. L. Michels

Tested on: Mathematica 14.0

Section 0

Results of Section 0 is used in Theorem 1 (see paper)

In[271]:=

```
(*Define the initial parameters*)
δ1 = Pi / 2;

(*Angles for the vertices*)
anglesRad = {
   {α1, β1, γ1, δ1}, (*Vertex 1*)
   {δ1, γ1, β1, α1}, (*Vertex 2*)
   {Pi - δ1, Pi - γ1, Pi - β1, Pi - α1}, (*Vertex 3*)
   {α1, β1, γ1, δ1}  (*Vertex 4*)
  };

(*Compute sigma values for each vertex*)
computeSigma[{α_, β_, γ_, δ_}] := (α + β + γ + δ) / 2;
sigmas = computeSigma /@ anglesRad;

(*Compute the differences for convenience*)
σ1 = sigmas[[1]] ;
```

```
ᾱ1 = σ1 - α1;
β̄1 = σ1 - β1;
γ̄1 = σ1 - γ1;
δ̄1 = σ1 - δ1;

(*Bricard's equation as defined in terms of angles and variables*)
BricardsEquation[{α_, β_, γ_, δ_}, σ_, x_, y_] := Module[
   {c22, c20, c02, c11, c00},
   c22 = Sin[σ - δ] Sin[σ - δ - β];
   c20 = Sin[σ - α] Sin[σ - α - β];
   c02 = Sin[σ - γ] Sin[σ - γ - β];
   c11 = -Sin[α] Sin[γ];
   c00 = Sin[σ] Sin[σ - β];
   c22 x^2 y^2 + c20 x^2 + c02 y^2 + 2 c11 x y + c00
  ];

(*Flexion 1*)
e0 = +1;
Z[t_] :=
  (-t * Sin[β1] + e0 * Sqrt[(t^2 * Sin[ᾱ1] * Sin[β̄1] + Sin[σ1] * Sin[ᾱ1 - β1]) *
        (Sin[γ̄1] * Sin[δ̄1] + t^2 * Sin[γ̄1 - β1] * Sin[δ̄1 - β1]) ]) /
   (Sin[δ̄1] * Sin[ᾱ1 - β1] + t^2 * Sin[ᾱ1] * Sin[δ̄1 - β1] );

W2[t_] := t;

U[t_] :=
  (t * Sin[β1] + e0 * Sqrt[(t^2 * Sin[ᾱ1] * Sin[β̄1] + Sin[σ1] * Sin[ᾱ1 - β1]) *
        (Sin[γ̄1] * Sin[δ̄1] + t^2 * Sin[γ̄1 - β1] * Sin[δ̄1 - β1]) ]) /
   (Sin[δ̄1] * Sin[ᾱ1 - β1] + t^2 * Sin[ᾱ1] * Sin[δ̄1 - β1] );

W1[t_] := e0 * Sqrt[(t^2 * Sin[ᾱ1] * Sin[β̄1] + Sin[σ1] * Sin[ᾱ1 - β1]) *
      (Sin[γ̄1] * Sin[δ̄1] + t^2 * Sin[γ̄1 - β1] * Sin[δ̄1 - β1]) ] /
    (Sin[γ̄1] * Sin[δ̄1] + t^2 * Sin[γ̄1 - β1] * Sin[δ̄1 - β1] );

(*Compute the expressions for Bricard's equations*)
P1 = BricardsEquation[anglesRad[[1]], sigmas[[1]], Z[t], W1[t]];
P2 = BricardsEquation[anglesRad[[2]], sigmas[[2]], Z[t], W2[t]];
P3 = BricardsEquation[anglesRad[[3]], sigmas[[3]], U[t], W2[t]];
P4 = BricardsEquation[anglesRad[[4]], sigmas[[4]], U[t], W1[t]];

(*Full simplification of the equations*)
simplifiedP1 = FullSimplify[P1];
simplifiedP2 = FullSimplify[P2];
simplifiedP3 = FullSimplify[P3];
simplifiedP4 = FullSimplify[P4];
```

```
(*LHS and RHS for the table*)
lhs = {"P₁(Z, W₁)", "P₂(Z, W₂)", "P₃(U, W₂)", "P₄(U, W₁)"};
rhs = {simplifiedP1, simplifiedP2, simplifiedP3, simplifiedP4};

(*Check if the expression simplifies to zero*)
isZeroQ[expr_] := TrueQ[FullSimplify[expr] === 0];

(*Create the table with a title manually inserted*)
title = "Flexion 1";
(*Create the table*)
table = TableForm[Table[{lhs⟦i⟧, rhs⟦i⟧, If[isZeroQ[rhs⟦i⟧], "✓", "✗"]},
    {i, Length[lhs]}], TableHeadings → {None, {"LHS", "RHS", "LHS = RHS?"}}];

(*Combine the title and the table*)
Column[{title, table}]



(*Flexion 2*)
ClearAll[Z, W2, U, W1, e0]
e0 = -1;
Z[t_] :=
  (-t * Sin[β1] + e0 * Sqrt[(t^2 * Sin[ᾱ1] * Sin[β̄1] + Sin[σ1] * Sin[ᾱ1 - β1]) *
        (Sin[γ̄1] * Sin[δ̄1] + t^2 * Sin[γ̄1 - β1] * Sin[δ̄1 - β1]) ]) /
   (Sin[δ̄1] * Sin[ᾱ1 - β1] + t^2 * Sin[ᾱ1] * Sin[δ̄1 - β1] );

W2[t_] := t;

U[t_] :=
  (t * Sin[β1] + e0 * Sqrt[(t^2 * Sin[ᾱ1] * Sin[β̄1] + Sin[σ1] * Sin[ᾱ1 - β1]) *
        (Sin[γ̄1] * Sin[δ̄1] + t^2 * Sin[γ̄1 - β1] * Sin[δ̄1 - β1]) ]) /
   (Sin[δ̄1] * Sin[ᾱ1 - β1] + t^2 * Sin[ᾱ1] * Sin[δ̄1 - β1] );

W1[t_] := e0 *  Sqrt[(t^2 * Sin[ᾱ1] * Sin[β̄1] + Sin[σ1] * Sin[ᾱ1 - β1]) *
      (Sin[γ̄1] * Sin[δ̄1] + t^2 * Sin[γ̄1 - β1] * Sin[δ̄1 - β1]) ] /
    (Sin[γ̄1] * Sin[δ̄1] + t^2 * Sin[γ̄1 - β1] * Sin[δ̄1 - β1] );

(*Compute the expressions for Bricard's equations*)
P1 = BricardsEquation[anglesRad⟦1⟧, sigmas⟦1⟧, Z[t], W1[t]];
P2 = BricardsEquation[anglesRad⟦2⟧, sigmas⟦2⟧, Z[t], W2[t]];
P3 = BricardsEquation[anglesRad⟦3⟧, sigmas⟦3⟧, U[t], W2[t]];
P4 = BricardsEquation[anglesRad⟦4⟧, sigmas⟦4⟧, U[t], W1[t]];

(*Full simplification of the equations*)
simplifiedP1 = FullSimplify[P1];
simplifiedP2 = FullSimplify[P2];
simplifiedP3 = FullSimplify[P3];
```

```
simplifiedP4 = FullSimplify[P4];

(*LHS and RHS for the table*)
lhs = {"P₁(Z, W₁)", "P₂(Z, W₂)", "P₃(U, W₂)", "P₄(U, W₁)"};
rhs = {simplifiedP1, simplifiedP2, simplifiedP3, simplifiedP4};

(*Check if the expression simplifies to zero*)
isZeroQ[expr_] := TrueQ[FullSimplify[expr] === 0];

(*Create the table with a title manually inserted*)
title = "Flexion 2";
(*Create the table*)
table = TableForm[Table[{lhs[[i]], rhs[[i]], If[isZeroQ[rhs[[i]]], "✓", "✗"]},
    {i, Length[lhs]}], TableHeadings → {None, {"LHS", "RHS", "LHS = RHS?"}}];

(*Combine the title and the table*)
Column[{title, table}]
```

Out[299]=

Flexion 1

LHS	RHS	LHS = RHS?
$P_1(Z, W_1)$	0	✓
$P_2(Z, W_2)$	0	✓
$P_3(U, W_2)$	0	✓
$P_4(U, W_1)$	0	✓

Out[319]=

Flexion 2

LHS	RHS	LHS = RHS?
$P_1(Z, W_1)$	0	✓
$P_2(Z, W_2)$	0	✓
$P_3(U, W_2)$	0	✓
$P_4(U, W_1)$	0	✓

Section 1

Results of Section 1 is used in Lemma 1 (see paper) and in Sections 2 and 3 of this file

```
(*Clear all previous definitions*)
ClearAll[α, β, γ, δ, σ, ᾱ, β̄, γ̄, δ̄, a, b, c, d, M]

(*Define the angles and variables*)
σ = (α + β + γ + δ) / 2;
ᾱ = σ - α;
β̄ = σ - β;
γ̄ = σ - γ;
δ̄ = σ - δ;

a = Sin[α] / Sin[ᾱ];
b = Sin[β] / Sin[β̄];
c = Sin[γ] / Sin[γ̄];
d = Sin[δ] / Sin[δ̄];
M = a * b * c * d;


(*Define the 7 identities*)
expr1 = 1 - a * b == Sin[σ] * Sin[ᾱ - β] / (Sin[ᾱ] * Sin[β̄]);
expr2 = 1 - b * c == Sin[σ] * Sin[γ̄ - β] / (Sin[γ̄] * Sin[β̄]);
expr3 = 1 - b * d == Sin[σ] * Sin[δ̄ - β] / (Sin[δ̄] * Sin[β̄]);
expr4 = c * d - 1 == Sin[σ] * Sin[ᾱ - β] / (Sin[γ̄] * Sin[δ̄]);
expr5 = a * d - 1 == Sin[σ] * Sin[γ̄ - β] / (Sin[ᾱ] * Sin[δ̄]);
expr6 = a * c - 1 == Sin[σ] * Sin[δ̄ - β] / (Sin[ᾱ] * Sin[γ̄]);
expr7 = 1 - M ==
   Sin[σ] * Sin[ᾱ - β] * Sin[γ̄ - β] * Sin[δ̄ - β] / (Sin[ᾱ] * Sin[β̄] * Sin[γ̄] * Sin[δ̄]);

(*Store the expressions and labels*)
lhs = {"1 - ab", "1 - bc", "1 - bd", "cd - 1", "ad - 1", "ac - 1", "1 - M"};
rhs =
  {"sin(σ)sin(ᾱ - β) / (sin(ᾱ)sin(β̄))", "sin(σ)sin(γ̄ - β) / (sin(β̄)sin(γ̄))",
   "sin(σ)sin(δ̄ - β) / (sin(β̄)sin(δ̄))", "sin(σ)sin(ᾱ - β) / (sin(γ̄)sin(δ̄)",
   "sin(σ)sin(γ̄ - β) / (sin(ᾱ)sin(δ̄))", "sin(σ)sin(δ̄ - β) / (sin(ᾱ)sin(γ̄))",
   "sin(σ)sin(ᾱ - β)sin(γ̄ - β)sin(δ̄ - β) / (sin(ᾱ)sin(β̄)sin(γ̄)sin(δ̄))"};

expressions = {expr1, expr2, expr3, expr4, expr5, expr6, expr7};

(*Function to test equality*)
isTrueQ[expr_] := TrueQ[FullSimplify[expr]]

(*Build the result table*)
TableForm[Table[{lhs[[i]], rhs[[i]], If[isTrueQ[expressions[[i]]], "✓", "✗"]},
  {i, Length[expressions]}],
 TableHeadings → {None, {"LHS", "RHS", "LHS = RHS?"}}]
```

Out[ ]//TableForm=

LHS	RHS	LI
1 - ab	sin(σ)sin(ᾱ - β) / (sin(ᾱ)sin(β̄))	✓
1 - bc	sin(σ)sin(γ̄ - β) / (sin(β̄)sin(γ̄))	✓
1 - bd	sin(σ)sin(δ̄ - β) / (sin(β̄)sin(δ̄))	✓
cd - 1	sin(σ)sin(ᾱ - β) / (sin(γ̄)sin(δ̄)	✓
ad - 1	sin(σ)sin(γ̄ - β) / (sin(ᾱ)sin(δ̄))	✓
ac - 1	sin(σ)sin(δ̄ - β) / (sin(ᾱ)sin(γ̄))	✓
1 - M	sin(σ)sin(ᾱ - β)sin(γ̄ - β)sin(δ̄ - β) / (sin(ᾱ)sin(β̄)sin(γ̄)sin(δ̄))	✓

Section 2

Results of Section 2 is used in the proof of Lemma 6 (see paper)

```
(*Clear all previous definitions*)
ClearAll[α, β, γ, δ, σ, ᾱ, β̄, γ̄, δ̄, u, x, y, z, d1, d2, d3, d4, d5,
 rhsCosAlpha, rhsCosBeta, rhsCosGamma, rhsCosDelta, rhsCosSigma]

(*Define assumptions for the angles*)
assumptions = {0 < α < Pi, 0 < β < Pi, 0 < γ < Pi,
   0 < δ < Pi, 0 < ᾱ < Pi, 0 < β̄ < Pi, 0 < γ̄ < Pi, 0 < δ̄ < Pi};

(*Define the angles and intermediate variables*)
σ = (α + β + γ + δ) / 2;
ᾱ = σ - α;
β̄ = σ - β;
γ̄ = σ - γ;
δ̄ = σ - δ;
ε = Abs[Sin[σ]] / Sin[σ];

(*Define Section 1 results*)
OneMinusM =
  Sin[σ] * Sin[ᾱ - β] * Sin[γ̄ - β] * Sin[δ̄ - β] / (Sin[ᾱ] * Sin[β̄] * Sin[γ̄] * Sin[δ̄]);
rMinusOne = Sin[σ] * Sin[γ̄ - β] / (Sin[ᾱ] * Sin[δ̄]);
sMinusOne = Sin[σ] * Sin[ᾱ - β] / (Sin[γ̄] * Sin[δ̄]);
fMinusOne = Sin[σ] * Sin[δ̄ - β] / (Sin[ᾱ] * Sin[γ̄]);

u = OneMinusM;
x = 1 / rMinusOne;
y = 1 / sMinusOne;
z = 1 / fMinusOne;

(*Define expressions in denominators*)
d1 = x * z * u * (1 + y) * (1 + u * y);
d2 = y * z * u * (1 + x) * (1 + u * x);
```

```
d3 = x * y * u * (1 + z) * (1 + u * z);
d4 = u * (1 + x) * (1 + y) * (1 + z) * (1 + u * x) * (1 + u * y) * (1 + u * z);
d5 = x * y * z * u^2 * (1 + x) * (1 + y) * (1 + z);

(*Define denominators*)
denAlpha = 2 * FullSimplify[Sqrt[d1], assumptions];
denBeta = 2 * FullSimplify[Sqrt[d4], assumptions];
denGamma = 2 * FullSimplify[Sqrt[d2], assumptions];
denDelta = 2 * FullSimplify[Sqrt[d3], assumptions];
denSigma = 2 * FullSimplify[Sqrt[d5], assumptions];

(*Define RHS expressions*)
rhsCosAlpha = FullSimplify[ε * (1 - y * z * u + x * z * u - x * y * u) / denAlpha] /.
   Cos[β] - Cos[α + γ + δ] :> 2 Sin[(α + β + γ + δ) / 2] Sin[(α - β + γ + δ) / 2];

rhsCosBeta =
  FullSimplify[ε * (u * (1 + x) * (1 + y) * (1 + z) + (1 + u * x) * (1 + u * y) * (1 + u * z) -
        u * x * y * z * (u - 1) ^ 2) / denBeta] /.
   Cos[α] - Cos[β + γ + δ] :> 2 Sin[(α + β + γ + δ) / 2] Sin[(-α + β + γ + δ) / 2];

rhsCosGamma = FullSimplify[ε * (1 + y * z * u - x * z * u - x * y * u) / denGamma] /.
   Cos[β] - Cos[α + γ + δ] :> 2 Sin[(α + β + γ + δ) / 2] Sin[(α - β + γ + δ) / 2];

rhsCosDelta = FullSimplify[ε * (1 - y * z * u - x * z * u + x * y * u) / denDelta] /.
   Cos[β] - Cos[α + γ + δ] :> 2 Sin[(α + β + γ + δ) / 2] Sin[(α - β + γ + δ) / 2];

rhsCosSigma = (1 - u * (x * y + x * z + y * z + 2 * x * y * z)) / denSigma;

(*Define equations to check*)
exprAlpha = Cos[α] == FullSimplify[rhsCosAlpha, assumptions];
exprBeta = Cos[β] == FullSimplify[rhsCosBeta, assumptions];
exprGamma = Cos[γ] == FullSimplify[rhsCosGamma, assumptions];
exprDelta = Cos[δ] == FullSimplify[rhsCosDelta, assumptions];
exprSigma = Cos[σ] == FullSimplify[rhsCosSigma, assumptions];

(*Build labeled table*)
lhs = {"cos(α)", "cos(β)", "cos(γ)", "cos(δ)", "cos(σ)"};
rhs = {"ε(1 - yzu + xzu - xyu) / (2√(xzu(1 + y)(1 + uy)))",
   "ε(u(1 + x)(1 + y)(1 + z) + (1 + ux)(1 + uy)(1 + uz) - uxyz(u -
     1)^2) / (2√(u(1 + x)(1 + y)(1 + z)(1 + ux)(1 + uy)(1 + uz)))",
   "ε(1 + yzu - xzu - xyu) / (2√(yzu(1 + x)(1 + ux)))",
   "ε(1 - yzu - xzu + xyu) / (2√(xyu(1 + z)(1 + uz)))",
   "(1 - u(xy + xz + yz + 2xyz)) / (2√(xyzu^2(1 + x)(1 + y)(1 + z)))"};
expressions = {exprAlpha, exprBeta, exprGamma, exprDelta, exprSigma};
```

```
isTrueQ[expr_] := TrueQ[FullSimplify[expr, assumptions]];

TableForm[Table[{lhs〚i〛, rhs〚i〛, If[isTrueQ[expressions〚i〛], "✓", "✗"]},
  {i, Length[expressions]}],
 TableHeadings → {None, {"LHS", "RHS", "LHS = RHS?"}}]


(*(*old version:check equalities*)
(*cos(α)*)
FullSimplify[
  (ε (Sin[σ - β] Sin[σ] + Sin[σ - α] Sin[σ - α - β] - Sin[σ - γ] Sin[σ - γ -
          β] - Sin[σ - δ] Sin[σ - δ - β])) / (2 Sin[γ] Sin[δ]) == ε Cos[α]]

 (*cos(β)*)
 FullSimplify[
  (ε (Sin[σ - α] Sin[σ] + Sin[σ - β] Sin[σ - α - β] +  Sin[σ - δ] Sin[σ - γ -
          β] + Sin[σ - γ] Sin[σ - δ - β])) / (2 Sin[γ] Sin[δ]) == ε Cos[β]]

 (*cos(γ)*)
 FullSimplify[
  (ε (Sin[σ - β] Sin[σ] - Sin[σ - α] Sin[σ - α - β] + Sin[σ - γ] Sin[σ - γ -
          β] - Sin[σ - δ] Sin[σ - δ - β])) / (2 Sin[α] Sin[δ]) == ε Cos[γ]]

 (*cos(δ)*)
 FullSimplify[
  (ε (Sin[σ - β] Sin[σ] - Sin[σ - α] Sin[σ - α - β] - Sin[σ - γ] Sin[σ - γ -
          β] + Sin[σ - δ] Sin[σ - δ - β])) / (2 Sin[α] Sin[γ]) == ε Cos[δ]]

 (*cos(σ)*)
 FullSimplify[(Sin[σ - β] Sin[σ] ^2 - Sin[σ - α] Sin[σ - α - β] Sin[σ] -
      Sin[σ - γ] Sin[σ - γ - β] Sin[σ] - Sin[σ - δ] Sin[σ - δ - β] Sin[σ] -
      2 Sin[σ - α] Sin[σ - γ] Sin[σ - δ]) /
    (2 Sin[α] Sin[γ] Sin[δ]) == Cos[σ]]*)
```

Out[∘]//TableForm=

LHS	RHS
cos(α)	ε(1 - yzu + xzu - xyu) / (2√(xzu(1 + y)(1 + uy)))
cos(β)	ε(u(1 + x)(1 + y)(1 + z) + (1 + ux)(1 + uy)(1 + uz) - uxyz(u - 1)^2) /
cos(γ)	ε(1 + yzu - xzu - xyu) / (2√(yzu(1 + x)(1 + ux)))
cos(δ)	ε(1 - yzu - xzu + xyu) / (2√(xyu(1 + z)(1 + uz)))
cos(σ)	(1 - u(xy + xz + yz + 2xyz)) / (2√(xyzu^2(1 + x)(1 + y)(1 + z)))

Section 3

Results of Section 3 is used in the proof of Lemma 7 (see paper)

In[∘]:=
```
(*Clear previous definitions*)
ClearAll[α, β, γ, δ, σ, ᾱ, β̄, γ̄, δ̄, ε, θ, A, ξ, η, lhs, rhs, expr, thetaIndex];
```

```
(*Define assumptions for the angles*)
assumptions = {0 < α < Pi, 0 < β < Pi, 0 < γ < Pi,
   0 < δ < Pi, 0 < ᾱ < Pi, 0 < β̄ < Pi, 0 < γ̄ < Pi, 0 < δ̄ < Pi};

(*---Definitions---*)
σ = (α + β + γ + δ) / 2;
ᾱ = σ - α;
β̄ = σ - β;
γ̄ = σ - γ;
δ̄ = σ - δ;
ε = Abs[Sin[σ]] / Sin[σ];

(*Define Section 1 results*)
OneMinusM =
  Sin[σ] * Sin[ᾱ - β] * Sin[γ̄ - β] * Sin[δ̄ - β] / (Sin[ᾱ] * Sin[β̄] * Sin[γ̄] * Sin[δ̄]);
rMinusOne = Sin[σ] * Sin[γ̄ - β] / (Sin[ᾱ] * Sin[δ̄]);
sMinusOne = Sin[σ] * Sin[ᾱ - β] / (Sin[γ̄] * Sin[δ̄]);
fMinusOne = Sin[σ] * Sin[δ̄ - β] / (Sin[ᾱ] * Sin[γ̄]);

u = OneMinusM;
x = 1 / rMinusOne;
y = 1 / sMinusOne;
z = 1 / fMinusOne;

(*Define expressions in denominators*)
dexp = x * y * u * (1 + z) * (1 + u * z);

(*Define denominators*)
den = 2 * FullSimplify[Sqrt[dexp], assumptions];

(*Cyclic index function*)
thetaIndex[k_] := Mod[k - 1, 4] + 1;

(*Definition of A[i,j]*)
A[i_, j_] /; 1 ≤ i ≤ 4 && 1 ≤ j ≤ 4 :=
  4 Cos[θ[thetaIndex[i]] / 2 + (Pi / 4) i j (j - 1) + (Pi / 2) j] ^2 *
   Cos[θ[thetaIndex[i - 1]] / 2 + (Pi / 4) (i - 1) j (j - 1) + (Pi / 2) j] ^2;
(*Table[A[i,j],{i,1,4},{j,1,4}];*)

(*Define ξ[i] and η[i]*)
ξ[i_?IntegerQ] /; 1 ≤ i ≤ 4 :=
  If[OddQ[i], θ[thetaIndex[i]], θ[thetaIndex[i - 1]]];
η[i_?IntegerQ] /; 1 ≤ i ≤ 4 := If[OddQ[i], θ[thetaIndex[i - 1]], θ[thetaIndex[i]]];
(*Table[{i,ξ[i],η[i]},{i,1,4}];*)
```

```
(*Define left-hand side and right-hand side expressions*)
lhs[i_Integer] := FullSimplify[
    (A[i, 1] + A[i, 2] * y * z * u + A[i, 3] * x * z * u + A[i, 4] * x * y * u) / den] /.
   Cos[β] - Cos[α + γ + δ] :> 2 Sin[(α + β + γ + δ) / 2] Sin[(α - β + γ + δ) / 2];

rhs[i_Integer] :=
  ε (Cos[β] - Cos[γ] (Cos[α] Cos[δ] + Cos[ξ[i]] Sin[α] Sin[δ]) - Cos[η[i]]
       Sin[γ] (Cos[α] Sin[δ] - Cos[ξ[i]] Sin[α] Cos[δ])) / (Sin[α] Sin[γ]);

expr[i_Integer] := FullSimplify[Equal[lhs[i], rhs[i]], assumptions];

(*---Build labeled result table---*)
lhsList =
  {"(A11 + A12y1z1u1 + A13x1z1u1 + A14x1y1u1) / (2√(x1y1u1(1 + z1)(1
     + u1z1))", "(A21 + A22y2z2u2 + A23x2z2u2 +
     A24x2y2u2) / (2√(x2y2u2(1 + z2)(1 + u2z2))",
   "(A31 + A32y3z3u3 + A33x3z3u3 + A34x3y3u3) / (2√(x3y3u3(1 +
     z3)(1 + u3z3))", "(A41 + A42y4z4u4 + A43x4z4u4
     + A44x4y4u4) / (2√(x4y4u4(1 + z4)(1 + u4z4))"};

rhsList = {"ε1((cos(β1) - cos(γ1)(cos(α1)cos(δ1) +
     cos(θ1)sin(α1)sin(δ1)) - cos(θ4)sin(γ1)(cos(α1)sin(δ1)
     - cos(θ1)sin(α1)cos(δ1))) / (sin(α1)sin(γ1))",
   "ε2((cos(β2) - cos(γ2)(cos(α2)cos(δ2) + cos(θ1)sin(α2)sin(δ2))
     - cos(θ2)sin(γ2)(cos(α2)sin(δ2) -
     cos(θ1)sin(α2)cos(δ2))) / (sin(α2)sin(γ2))",
   "ε3((cos(β3) - cos(γ3)(cos(α3)cos(δ3) + cos(θ3)sin(α3)sin(δ3))
     - cos(θ2)sin(γ3)(cos(α3)sin(δ3) -
     cos(θ3)sin(α3)cos(δ3))) / (sin(α3)sin(γ3))",
   "ε4((cos(β4) - cos(γ4)(cos(α4)cos(δ4) + cos(θ3)sin(α4)sin(δ4))
     - cos(θ4)sin(γ4)(cos(α4)sin(δ4) -
     cos(θ3)sin(α4)cos(δ4))) / (sin(α4)sin(γ4))"};

results = Table[If[TrueQ[expr[i]], "✓", "✗"], {i, 1, 4}];

TableForm[Table[{i, lhsList[[i]], rhsList[[i]], results[[i]]}, {i, 1, 4}],
 TableHeadings -> {None, {"i", "LHS", "RHS", "LHS = RHS?"}},
 TableAlignments -> Left]
```

Out[∘]//TableForm=

i	LHS
1	$(A_{11} + A_{12}y_1z_1u_1 + A_{13}x_1z_1u_1 + A_{14}x_1y_1u_1) / (2\sqrt{}(x_1y_1u_1(1 + z_1)(1 + u_1z_1))$
2	$(A_{21} + A_{22}y_2z_2u_2 + A_{23}x_2z_2u_2 + A_{24}x_2y_2u_2) / (2\sqrt{}(x_2y_2u_2(1 + z_2)(1 + u_2z_2))$
3	$(A_{31} + A_{32}y_3z_3u_3 + A_{33}x_3z_3u_3 + A_{34}x_3y_3u_3) / (2\sqrt{}(x_3y_3u_3(1 + z_3)(1 + u_3z_3))$
4	$(A_{41} + A_{42}y_4z_4u_4 + A_{43}x_4z_4u_4 + A_{44}x_4y_4u_4) / (2\sqrt{}(x_4y_4u_4(1 + z_4)(1 + u_4z_4))$