A Cross-Ratio Criterion for the Circular Position of Points in Space

Hans-Peter Schröcker
July 2010

```
> restart: with(LinearAlgebra):
```

We want to show that four points a, b, c, and $d \in \mathbb{R}^3 \subset \mathbb{H}$ (the division ring of quaternions) lie on a circle or a straight line if and only if the expression

$$(*) \qquad (a-b)\cdot(b-c)^{-1}\cdot(c-d)\cdot(d-a)$$

is real. The points in $\mathbb{R}^3$ are embedded as $(0, a)$, $(0, b)$ etc. into $\mathbb{H}$.

Equation $(*)$ is a cross-ratio-like expression. In this context, we simply call it *cross-ratio.*

Clearly, the cross-ratio $(*)$ is invariant with respect to translations ($x \mapsto x + t$) and scaling ($x \mapsto \lambda x$). Therefore, it is no loss of generality to assume that a, b, c, and d are contained in the unit sphere $S^2 \subset \mathbb{R}^3 \subset \mathbb{H}$. We compute the cross-ratio for this special case.

Procedure for quaternion multiplication:

```
> qm := proc(p, q)
    return Vector([p[1]*q[1]-p[2]*q[2]-p[3]*q[3]-p[4]*q[4],
                   p[1]*q[2]+p[2]*q[1]+p[3]*q[4]-p[4]*q[3],
                   p[1]*q[3]-p[2]*q[4]+p[3]*q[1]+p[4]*q[2],
                   p[1]*q[4]+p[2]*q[3]-p[3]*q[2]+p[4]*q[1]]):
  end proc:
```

Procedure for conjugate quaternion. Note that conjugation equals inversion for quaternions on S^2.

```
> qc := proc(p) return Vector([p[1], -p[2], -p[3], -p[4]]): end
  proc:
```

Now we make use of a facts which are well-known or easy to see:

1. The complex plane can be embedded into the quaternion division ring $(\mathbb{H}, +, \cdot)$ via $(x + \mathrm{i}\cdot y) \hookrightarrow (x, y, 0, 0)$.
2. Expression $(*)$ is real for concircular or collinear points in the complex plane.
3. Stereographic projection maps concircular points of S^2 onto concircular or collinear point in $\mathbb{R}^2$.

Thus, we may start with four arbitrary points in $\mathbb{R}^2$, compute their (complex) cross-ratio and compare it with the cross-ratio of their stereographic pre-images on S^2.

```
> a := Vector([xa, ya, 0, 0]):
  b := Vector([xb, yb, 0, 0]):
  c := Vector([xc, yc, 0, 0]):
  d := Vector([xd, yd, 0, 0]):
```

```
> stereographicPreImage := proc(X)
    local d:
    d := 1+X[1]^2 + X[2]^2:
```

```
    return 1/d * Vector([0, 2*X[1], 2*X[2], -1 + X[1]^2 + X[2]^2])
  :
  end proc:
> A := stereographicPreImage(a):
  B := stereographicPreImage(b):
  C := stereographicPreImage(c):
  unprotect(D):
  D := stereographicPreImage(d):
> q := qm(qm(a-b, qc(b-c)), qm(c-d, qc(d-a))):
  Q := qm(qm(A-B, qc(B-C)), qm(C-D, qc(D-A))):
```

Now we discuss the vanishing of the second component of q and the second, third, and fourth component of Q.

```
> q2 := numer(normal(q[2])):
> Q2 := map(numer, map(normal, Q)):
> simplify(1/(16*q2) * Q2[2..4]);
```

$$\begin{bmatrix} -2\,xa \\ -2\,ya \\ 1 - xa^2 - ya^2 \end{bmatrix} \quad \textbf{(1)}$$

We see that q_2 is a factor of each of the relevant components of Q_2. Its vanishing implies $Q \in \mathbb{R}$. Conversely, if $Q \in \mathbb{R}$, q_2 must vanish because not all components of (1) can vanish simultaneously.