curvature

November 18, 2019

```
In [1]: from sympy import *
```

```
In [2]: init_printing()
```

```
In [3]: a, r, u, v = symbols('a r u v', real=True)
```

```
In [4]: # x = Matrix([
        #     (a + r * cos(u)) * cos(v),
        #     (a + r * cos(u)) * sin(v),
        #     r * sin(u)
        # ])
        x = Matrix([
            u - power.Pow(u, 3)/3 + u * power.Pow(v, 2),
            v - power.Pow(v, 3)/3 + v * power.Pow(u, 2),
            power.Pow(u, 2) - power.Pow(v, 2)
        ])
```

```
In [5]: x_u = diff(x, u)
        x_v = diff(x, v)
        x_uu = diff(x_u, u)
        x_uv = diff(x_u, v)
        x_vv = diff(x_v, v)
        N = x_u.cross(x_v).normalized()
        N.simplify()
```

```
In [6]: x_u, x_v, x_uu, x_uv, x_vv
```

Out[6]:

$$\left(\begin{bmatrix} -u^2 + v^2 + 1 \\ 2uv \\ 2u \end{bmatrix}, \begin{bmatrix} 2uv \\ u^2 - v^2 + 1 \\ -2v \end{bmatrix}, \begin{bmatrix} -2u \\ 2v \\ 2 \end{bmatrix}, \begin{bmatrix} 2v \\ 2u \\ 0 \end{bmatrix}, \begin{bmatrix} 2u \\ -2v \\ -2 \end{bmatrix} \right)$$

```
In [7]: def simp_dot(a, b):
            expr = a.dot(b)
            return expr.simplify().trigsimp().factor().simplify()
        E = simp_dot(x_u, x_u)
        F = simp_dot(x_u, x_v)
        G = simp_dot(x_v, x_v)
        e = simp_dot(N, x_uu)
        f = simp_dot(N, x_uv)
        g = simp_dot(N, x_vv)
```

In [8]:
```
E
```

Out[8]:
$$\left(u^2+v^2+1\right)^2$$

In [9]:
```
F
```

Out[9]:
$$0$$

In [10]:
```
G
```

Out[10]:
$$\left(u^2+v^2+1\right)^2$$

In [11]:
```
e
```

Out[11]:
$$2$$

In [12]:
```
f
```

Out[12]:
$$0$$

In [13]:
```
g
```

Out[13]:
$$-2$$

1 The curvatures are

In [16]:
```
gauss_curvature = (e*g - power.Pow(f, 2))/(E*G - power.Pow(F, 2)).simplify()
gauss_curvature
```

Out[16]:
$$-\frac{4}{\left(u^2+v^2+1\right)^4}$$

In [17]:
```
mean_curvature = .5 * (e * G - 2 * f * F + g * E)/(E * G - power.Pow(F, 2))
```

In [18]:
```
mean_curvature.simplify()
```

Out[18]:
$$0$$

In [19]:
```
the_root = power.Pow(power.Pow(mean_curvature, 2) - gauss_curvature, .5)
k1 = (mean_curvature + the_root).simplify()
k2 = (mean_curvature - the_root).simplify()
k1, k2
```

Out[19]:
$$\left(\frac{2.0}{\left(u^2+v^2+1\right)^{2.0}},\ -\frac{2.0}{\left(u^2+v^2+1\right)^{2.0}}\right)$$