Lunghezza di un arco di curva regolare

Marcello Colozzo - http://www.extrabyte.info

Sia data la curva regolare di rappresentazione parametrica

$x = x(t)$, $y = y(t)$, $z = z(t)$ con $t \in [a, b]$. L' ascissa curvilinea è data da $s(t) = \pm \int_{t_0}^{t} H(\tau)\, d\tau$,
dove t_0 è il valore del parametro corrispondente all' origine degli archi, mentre la funzione integranda è

$H(t) = +\sqrt{x'(t)^2 + y'(t)^2 + z'(t)^{\wedge}2}$

Nell'espressione di $s(t)$ va preso il segno $(+)$ se la curva è orientata nel verso delle t crescenti. Viceversa, va preso il segno $(-)$. *Mathematica* permette di calcolare con uno specifico comando la funzione $H(t)$. Innanzitutto carichiamo il package :

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

Il comando in question è **`ArcLengthFactor`**

```
In[2]:= ? ArcLengthFactor
```

ArcLengthFactor[$\{f_1, f_2, f_3\}$, t] gives the derivative of the arc length of the curve described by the parametrized curve coordinates $\{f_1, f_2, f_3\}$ with respect to the parameter t in the default coordinate system.
ArcLengthFactor[$\{f_1, f_2, f_3\}$, t, *coordsys*] gives the derivative of the arc length of a curve in the coordinate system *coordsys*. ≫

Ad esempio, consideriamo la curva:

```
In[3]:= x[t_] := 2 Cos[t]; y[t_] := 2 Sin[t]; z[t_] := 3/π t
```

Qui l'intervallo base è $[0,\pi]$.

```
In[4]:= Dx[t_] = ∂t x[t]; Dy[t_] = ∂t y[t]; Dz[t_] = ∂t z[t];

In[5]:= H[t_] = √(Dx[t]^2 + Dy[t]^2 + Dz[t]^2) // Simplify
```

Out[5]= $\dfrac{\sqrt{9 + 4\pi^2}}{\pi}$

Orientando la curva nel verso delle t crescenti :

```
In[6]:= s[t_] = Integrate[
          H[τ],
          {τ, 0, t},
          Assumptions → t ≥ 0
         ]
```

Out[6]= $\dfrac{\sqrt{9 + 4\pi^2}\; t}{\pi}$

La lunghezza è:

```
In[7]:= s[π]
```

Out[7]= $\sqrt{9 + 4\pi^2}$

```
In[8]:= Clear[x, y, z, s, H]
```

Nel caso appena visto, i calcoli sono semplici e non vale la pena di utilizzare il comando **`ArcLengthFactor`**. Consideriamo allora un esempio più complicato:

In[9]:=
```
x[t_] := Sinh[t]^4*Cos[t]; y[t_] := Exp[-t^2] Sin[2 t]; z[t_] := t^2 + 1
```

con intervallo base [0,1.8]

In[10]:=
```
curva = {x[t], y[t], z[t]}
```

Out[10]= $\left\{\text{Cos}[t]\,\text{Sinh}[t]^4,\ e^{-t^2}\,\text{Sin}[2\,t],\ 1+t^2\right\}$

In[11]:=
```
H[t_] = ArcLengthFactor[curva, t] // Simplify
```

Out[11]= $\sqrt{4\,t^2 + 4\,e^{-2\,t^2}\,(\text{Cos}[2\,t] - t\,\text{Sin}[2\,t])^2 + \text{Sinh}[t]^6\,(-4\,\text{Cos}[t]\,\text{Cosh}[t] + \text{Sin}[t]\,\text{Sinh}[t])^2}$

L'ascissa curvilinea

In[12]:=
```
s[t_] = Integrate[
  H[τ],
  {τ, 0, t},
  Assumptions → t ≥ 0
 ]
```

Out[32]= $Aborted

che non riesce a determinare in forma chiusa. Tentiamo numericamente:

In[12]:=
```
s[t_] := NIntegrate[
  H[τ],
  {τ, 0, t}
 ]
```

In[13]:=
```
s[0.2]
```

Out[13]= 0.377158

La lunghezza della curva:

In[14]:=
```
s[1]
```

Out[14]= 2.0418

Plottiamo la curva:

In[15]:=
```
plot = ParametricPlot3D[
  curva,
  {t, 0, 1.8},
  Boxed → False,
  AxesLabel →
   {
    Style["x", Medium, Red],
    Style["y", Medium, Blue],
    Style["z", Medium, Blue]
   },
  TicksStyle → Magenta,
  PlotStyle → {Blue, Thickness[0.004]}
 ]
```

Out[15]=