VISUAL AND INTERACTIVE PARALLEL TRANSPORT ON SURFACES WITH MATHEMATICA

Shuichi Yukita
Faculty of Computer and Information Sciences, Hosei University (JAPAN)

Abstract

In some areas of computer graphics, computer vision, and data compression, familiarity with classical differential geometry is required to develop efficient algorithms. Learning differential geometry involves considerable amount of calculations with pencil and paper, and yet gaining intuitive understanding is often a long way off. In engineering department education, we naturally seek an alternative way to attain the goal, which is the main theme of this paper. We present an interactive visualizer toolkit for developing teaching and self-learning materials in differential geometry of surfaces focusing on parallel transport targeting science and engineering department students. We give also present sample visualizers for the curvedness of surfaces.

Keywords: Interactive visualization, computer graphics, classical differential geometry, parallel transport, surfaces, Mathematica, mathematics laboratory.

1 INTRODUCTION

In some areas of shape modeling, computer vision, and data compression, familiarity with classical differential geometry is required to understand widely used algorithms. Ohtake et al.([1]) proposed a technique for implicit surface representation based on partition of unity, a very basic but considered purely theoretical notion in differential geometry. Watanebe et al.([2]) studied an approximation of Gaussian and the mean curvatures on the densely triangulated surfaces to detect local features of the surfaces. Yago et al.([3]) showed that the geometry of normal vector fields provides an efficient algorithm for smoothing noisy 3D shapes. To be a good user of these techniques, or to further advance technology in those fields, we have to be familiar with basic notions in differential geometry. So, it is natural to include differential geometry courses in the curricula of engineering departments.

Learning differential geometry involves considerable amount of calculations with pencil and paper, and yet gaining intuitive understanding is often a long way off. In engineering department education, we naturally seek an alternative way to attain the goal, which is the main theme of this paper.

Curvedness of surfaces appears in various ways. A good reference is Struik book([4]). We can easily recall that the sum of interior angles of a geodesic triangle on a sphere is greater than 180 degree, see Fig. 1, which means the sphere has a positive Gaussian curvature. Curvedness can also be measured by parallel translation of tangent vectors. Parallel transport or the Levi-Civita connection is one of the basic concepts in classical differential geometry. In a curved space, a vector transported parallel from the initial point to the destination along two different paths may result in different vectors at the destination point. An example of a parallel transport is shown in Fig. 2. The amount of this
difference depends on the curvedness of the space and the choice of paths. We present an interactive visualizer toolkit for developing teaching and self-learning materials in differential geometry of surfaces focusing on parallel transport targeting science and engineering department students. We also present sample interactive visualizers for the curvedness of surfaces.

Understanding parallel transport in a curved space is not straightforward for undergraduate learners since they have to solve differential equations of second order with initial conditions or boundary conditions according to the situations they are to tackle. Visualization of the effect of curvedness is much needed. See the author's related work concerning Gaussian curvature and topology in ([5]). The result of parallel transport varies according to the path. So, visualization should incorporate this variation of the result caused by the change of a path. Preferably further, the effect should appear promptly and smoothly when the user changes the configuration of a path continuously.

Our toolkit is designed to support teachers and course material developers to attain the goals above. It is built on top of Mathematica\textsuperscript{TM}, a widely used computer algebra system. The toolkit consists of a set of library functions and accompanied by a variety of samples for developing interactive visualizers.

The toolkit makes full use of 3D graphics and depends heavily on the dynamic object technology of Mathematica, not to mention its symbolic computation power. The toolkit also depends on the advanced numerical solution technology that provides numerical solutions of differential equations in the form of interpolation functions, which enables us to treat the numerical solutions not as an array of real numbers but as a normal function; we can differentiate them, plot them, and further compose them. We make full use of this feature.

The toolkit is accompanied by a set of instructional design patterns with which we can not only effectively develop teaching materials but also let students write their own code in hands-on laboratory sessions. Through the laboratory sessions, students gain deep insight both in mathematics and computing science.

This paper is organized as follows. In sections 2, 3, and 5, we describe patterns of instructions in class that demonstrate the effectiveness of our methods. In the last section, we conclude this paper.

2 ORTHONORMAL FRAMES AND PROJECTION TO THE TANGENT PLANE

Suppose we have a smooth surface in the Euclidean 3-space $\mathbb{R}^3$. And further suppose that students are requested to consider themselves to be an inhabitant that lives inside the two dimensional universe of their own, in this particular case, a surface in the 3-space. A presented theme for the students is how to measure changes of velocity in a curved space. We can do it fully intrinsically, namely without relying on the fact that the two dimensional universe is embedded in an ambient space $\mathbb{R}^3$, which belongs to another on-going project of ours. In this paper, we make full use of the embedding, which we may call an extrinsic geometry. A rough idea that we first differentiate any kind of geometrical quantities, subsequently apply orthogonal projection to the tangent space, and then we get what we wanted to calculate, namely the covariant derivatives.

Students are instructed to study a surface, for example, given in the parametric expression as follows.

\[ q(u, v) := (u, v, u^2 - v^2) \]

This surface is a saddle surface. See Fig. 3.

Students are first instructed to calculate an orthonormal basis of the tangent plane at each point via the Gram-Schmidt process. The most natural choice of linearly independent tangent vectors is

\[ \frac{\partial q}{\partial u}, \frac{\partial q}{\partial v} \]

After the Gram-Schmidt process, students are led to the following definitions of an orthonormal basis of the tangent space.

\[
\begin{align*}
 f_1(u, v) & := \left\{ \frac{1}{\sqrt{1 + 4u^2}}, \theta, \frac{2u}{\sqrt{1 + 4u^2}} \right\} \\
 f_2(u, v) & := \left\{ \frac{4uv}{\sqrt{1 + 4u^2}}, \frac{\sqrt{1 + 4u^2}}{1 + 4u^2 + 4v^2}, -\frac{2v}{\sqrt{1 + 4u^2 + 4v^2}} \right\}
\end{align*}
\]

By taking cross product of $f_1$ and $f_2$, namely
students find a unit normal vector

\[
f_3[u, v] := \left\{ \begin{array}{ll}
-\frac{2u}{\sqrt{1+4u^2+4v^2}}, & \\
\frac{2v}{\sqrt{1+4u^2+4v^2}}, & \\
\frac{1}{\sqrt{1+4u^2+4v^2}} & 
\end{array} \right. 
\]

Figure 3. A saddle surface

Students are instructed to move around the frame on the surface. The code suggested is:

\[
\text{Manipulate[}
\text{Show[}
\{g0,}
\text{Graphics3D[}
\{Red, Arrow[{q[u, v], q[u, v] + f1[u, v]}],}
\text{Green, Arrow[{q[u, v], q[u, v] + f2[u, v]}],}
\text{Blue, Arrow[{q[u, v], q[u, v] + f3[u, v]}]}\}
\},}
\text{PlotRange -> {{-2, 2}, {-2, 2}, {-2, 2}}},
\{u, -1, 1\}, \{v, -1, 1\},
\text{SaveDefinitions -> True}
\]
\]

Its output is in Fig. 4. Alternatively, or more intuitively, some students may do is as

\[
\text{DynamicModule[}\{pt = \{0, 0\}\},]
\]

\[
\text{Dynamic[}
\text{Show[}
\{g0,}
\text{Graphics3D[}
\{Red, Arrow[{q @@ pt, q @@ pt + f1 @@ pt}],}
\text{Green, Arrow[{q @@ pt, q @@ pt + f2 @@ pt}],}
\text{Blue, Arrow[{q @@ pt, q @@ pt + f3 @@ pt}]}}\}
\},}
\text{PlotRange -> {{-2, 2}, {-2, 2}, {-2, 2}}},
\text{Slider2D[Dynamic[pt], {{-1, -1}, {1, 1}}]}\]
\]

\text{SaveDefinitions -> True}
Next, students are instructed to seek for the projection operator $\text{prj}[u, v]$ that maps $f_1$ to $f_1$, $f_2$ to $f_2$, and $f_3$ to the null vector. The matrix representation of this operator is easily obtained (at least, for math department students without help, for engineering students with a little bit help) as follows. The code

\[
\text{Transpose}[\{f_1[u, v], f_2[u, v], \{0, 0, 0\}\}].\text{Inverse}[\
\text{Transpose}[\{f_1[u, v], f_2[u, v], f_3[u, v]\}])
\]

produces the result to be used in the following definition.

At this occasion, the instructor should find a good intervention point. Students are instructed to test if $\text{prj}$ is really an orthogonal projection. The suggested tests are:

\[
\text{Transpose}[\text{prj}[u, v]] == \text{prj}[u, v] \\
\text{prj}[u, v].\text{prj}[u, v] == \text{prj}[u, v] // \text{Simplify}
\]

Both Boolean expressions should evaluate to True.

3 GEODESICS

In this section, students are supposed to continue studying the same saddle surface given in the previous section.

3.1 Differential equations for geodesics

Let $(u_1, v_1)$ and $(u_2, v_2)$ be two points on the parameter plane. Students are instructed to find a geodesic on the surface with the metric induced by the ambient space $\mathbb{R}^3$. We consider a parametric curve $(u, v) = (f[t], g[t])$ with unknown functions $f[t]$ and $g[t]$. The corresponding curve on the surface is given as $q[f[t], g[t]]$ or $q @@\{f[t], g[t]\}$. Students are instructed to calculate covariant derivative of the curve with respect to $t$ twice. Notice that at the first differentiation, there is no need for taking projection since the result should lie in the tangent plane. However, it is a good idea for the instructor to suggest students to test if the following two calculations give the same result:

\[
\text{prj}[f[t], g[t]].D[q[f[t], g[t]], t, t] // \text{Simplify}
\]
and
\[ \text{prj}[f[t], g[t]].(D[\text{prj}[f[t], g[t]], f[t], t], t) + D[q[f[t], g[t]], g[t], t], t) \] // Simplify.

The result is
\[ \frac{4 f[t] (f[t]^2 - g[t]^2) + 4 f[t]^2 f'[t] - [1 + 4 g[t]^2] f'[t]}{1 + 4 f[t]^2 + 4 g[t]^2}, \quad \frac{-4 g[t] (f[t]^2 - g[t]^2) - [1 + 4 f[t]^2] g'[t] - 4 g[t]^2 g''[t]}{1 + 4 f[t]^2 + 4 g[t]^2}, \]
\[ \frac{2 (f[t]^2 - g[t]^2) g'[t] + 4 (f[t]^2 + g[t]^2) f'[t]}{1 + 4 f[t]^2 + 4 g[t]^2}, \quad \frac{1 + 4 f[t]^2 + 4 g[t]^2}{1 + 4 f[t]^2 + 4 g[t]^2}. \]

To obtain the second order differential equations of the geodesic, we must equate the above vector with the null vector. It is a straight forward task, but students are strongly encouraged to further transform the equations into a kind of principal form. In other words, students are suggested to solve the equations algebraically with respect to the leading terms \( f''[t] \) and \( g''[t] \), which enables Mathematica to efficiently solve the differential equations. The code for performing this task is as follows.

\[
\text{sol} = \text{Solve}\left[ \left\{ \frac{4 f[t] (f[t]^2 - g[t]^2) + 4 f[t]^2 f'[t] - [1 + 4 g[t]^2] f'[t]}{1 + 4 f[t]^2 + 4 g[t]^2}, \quad \frac{-4 g[t] (f[t]^2 - g[t]^2) - [1 + 4 f[t]^2] g'[t] - 4 g[t]^2 g''[t]}{1 + 4 f[t]^2 + 4 g[t]^2}, \right. \\
\left. \frac{2 (f[t]^2 - g[t]^2) g'[t] + 4 (f[t]^2 + g[t]^2) f'[t]}{1 + 4 f[t]^2 + 4 g[t]^2} \right\} = \{0, 0, 0, \{f''[t], g''[t]\}\} \right]
\]

This code produces the following algebraic solution.
\[ \{f''[t] = \frac{4 f[t] (f[t]^2 - g[t]^2)}{1 + 4 f[t]^2 + 4 g[t]^2}, g''[t] = \frac{4 g[t] (f[t]^2 - g[t]^2)}{1 + 4 f[t]^2 + 4 g[t]^2}\} \]

Students seek for the right hand sides of the differential equations to solve as follows.
\[ \{f''[t], g''[t]\} /. \text{sol}[1] \]

The result is:
\[ \left\{ -\frac{4 f[t] (f[t]^2 - g[t]^2)}{1 + 4 f[t]^2 + 4 g[t]^2}, \quad \frac{4 g[t] (f[t]^2 - g[t]^2)}{1 + 4 f[t]^2 + 4 g[t]^2} \right\}. \]

The differential equations are shown in a vector form as follows.
\[ \text{deq} = \{f''[t], g''[t]\} = \left\{ \frac{4 f[t] (f[t]^2 - g[t]^2)}{1 + 4 f[t]^2 + 4 g[t]^2}, \quad \frac{4 g[t] (f[t]^2 - g[t]^2)}{1 + 4 f[t]^2 + 4 g[t]^2} \right\} \]

### 3.2 Numerical solutions for geodesics

Students are instructed to solve the differential equations with an advanced numerical facility of Mathematica, using the \text{NDSolve} function that returns the numerical solutions in a form of spline functions. For example, a suggested code
\[ \text{ndsol} = \text{NDSolve}[(\text{deq}, \quad (f[0], g[0]) == (0, 0), \quad (f[1], g[1]) == (1, 1)), \{f, g\}, \{t, 0, 1\}] \]

produces the following result.
\[
\left\{ \{f \rightarrow \text{InterpolatingFunction}[[0, 1]], g \rightarrow \text{InterpolatingFunction}[[0, 1]]\}\right\}
\]

Since the solutions are given in a spline function, we can further differentiate or parametrically plot these quantities seamlessly. This is the advantage of the spline function numerical method. Students can benefit from this advanced numerical technology provided by Mathematica. The obtained geodesic is shown in Fig. 6.
In this subsection, students are supposed to generate an interactive geodesic animation. Since solving differential equations real time consumes excess amount of computing power, we have to cache the solutions and their 3D graphics data. Students are instructed to use the following techniques.

First, we fix one of the end points and let the other point move freely in some domain. We continue our study with the example in the previous subsection. Students are instructed to move the free point in a square in the \( u-v \) parameter space with a 10-by-10 mesh. The free point can take the position at the vertices of the mesh. The fixed point is placed at one of the corners. Students are then instructed to calculate the solutions of the geodesic equations and store the results in a table as follows.

```math
\text{soltab0 = Table[
  \text{NDSolve[}
    \{deq,
    \{f[0], g[0]\} == \{0, 0\},
    \{f[1], g[1]\} == \{i/10, j/10\},
    \{f, g\}, \{t, 0, 1\}],
    \{i, 10\}, \{j, 10\}\};
  \text{gtab0 = Table[
    \text{ParametricPlot3D[q @@ \{f[t], g[t]\} /. soltab0[[i, j, 1]], \{t, 0, 1\}],
    \{i, 10\}, \{j, 10\}\};
\text{With these stored data, students can create an interactive graphics.}
\text{Manipulate[}
  \text{Show[\{g0, gtab0[[i, j]]\}],
  \{i, 1, 10, 1\}, \{j, 1, 10, 1\},
  \text{SaveDefinitions -> True}]}
```
This code produces a manipulator shown in Fig. 7. Alternatively, the following code produces the same effect with a 2D slider, which may increase interactivity. The captured image is shown in Fig. 8.

```
DynamicModule[{pt = {0, 0}},
  {Dynamic[
    Show[
      g0,
      gtab0[[(Floor[10*pt[[1]]] + 1), (Floor[10*pt[[2]]] + 1)]] ]],
    Slider2D[Dynamic[pt], {{0, 0}, {0.9, 0.9}}]
  },
  SaveDefinitions -> True
]
```

Figure 8. Moving a geodesic with 2D slider

4 PARALLEL TRANSPORT

In this section, students are supposed to tackle with parallel transport. For simplicity, we will fix a square in the u-v parameter plane and move a tangent vector along the corresponding closed path on the saddle surface.

4.1 Specifying a close path

A square in the u-v parameter plane is given as follows.

```
L1[t_] := {t, 0} (* Case 0 <= t <= 1 *)
L2[t_] := {1, t - 1} (* Case 1 <= t <= 2 *)
L3[t_] := {3 - t, 1} (* Case 2 <= t <= 3 *)
L4[t_] := {0, 4 - t} (* Case 3 <= t <= 4 *)
```

```
loop[t_] := Which[
  0 <= t <= 1, L1[t],
  1 <= t <= 2, L2[t],
  2 <= t <= 3, L3[t],
  3 <= t <= 4, L4[t]
]
```

The corresponding curve on the saddle surface is given by the following code.

```
Show[g0, ParametricPlot3D[q @@ loop[t], {t, 0, 4}]]
```

The graphics output is shown in Fig. 9.
4.2 Differential equations for parallel transport

Our next task is to solve the differential equations of the geodesics. For this purpose, students are instructed to prepare a vector field expression with two unknown functions \( a[t] \) and \( b[t] \). Namely,

\[
\begin{align*}
vq_{1}[t] & := a[t]*(f1 @@ L1[t]) + b[t]*(f2 @@ L1[t]) \\
vq_{2}[t] & := a[t]*(f1 @@ L2[t]) + b[t]*(f2 @@ L2[t]) \\
vq_{3}[t] & := a[t]*(f1 @@ L3[t]) + b[t]*(f2 @@ L3[t]) \\
vq_{4}[t] & := a[t]*(f1 @@ L4[t]) + b[t]*(f2 @@ L4[t])
\end{align*}
\]

Students are instructed to set up differential equations, to solve them, and to visualize the results. In the first segment of the closed curve, the covariant derivative of the vector field is calculated by the code

\[
(prj @@ L1[t]).D[vq_{1}[t], t] // Simplify
\]

The result is

\[
\left\{ \frac{a'[t]}{\sqrt{1-4 t^2}}, \frac{b'[t]}{\sqrt{1-4 t^2}}, \frac{2 t a'[t]}{\sqrt{1-4 t^2}} \right\}.
\]

If we equate this quantity with the null vector, we get the differential equation for the parallel transport. Students should notice that the solutions are constant functions. Likewise, the situations are the same for the second and the last segments. The only exception is the third segment. We have to equate the following quantity with the null vector.

\[
(prj @@ L3[t]).D[vq_{3}[t], t] // Simplify
\]

If the initial conditions are given as \( a[0]=1 \) and \( b[0]=0 \), then the final result is obtained by the code below.

\[
\text{sol1 = NSolve}\left[ \left\{ \frac{4 b[t]}{(37 - 24 t + 4 t^2) \sqrt{41 - 24 t + 4 t^2}}, \frac{4 a[t]}{(37 - 24 t + 4 t^2) \sqrt{41 - 24 t + 4 t^2}}, \frac{2 t a'[t]}{\sqrt{1-4 t^2}} \right\}, \{a[2], b[2]\} == \{1, 0\}, \{a[0], b[0]\} == \{1, 0\} \right] \]

If the initial conditions are given as \( a[0]=0 \) and \( b[0]=1 \), then the final result is obtained by the code below.

\[
\text{sol2 = NSolve}\left[ \left\{ \frac{4 b[t]}{(37 - 24 t + 4 t^2) \sqrt{41 - 24 t + 4 t^2}}, \frac{4 a[t]}{(37 - 24 t + 4 t^2) \sqrt{41 - 24 t + 4 t^2}}, \frac{2 t a'[t]}{\sqrt{1-4 t^2}} \right\}, \{a[2], b[2]\} == \{0, 1\}, \{a[0], b[0]\} == \{0, 1\} \right] \]
Students are instructed to combine all the solutions to produce an interactive graphics as in Fig. 10.

4.3 Positive and negative curvatures

Fig. 11 shows comparison between positive and negative curvature cases. All the captured images are pairs of a saddle surface and a sphere, having negative and positive curvatures, respectively.

Students are suggested to do such calculation and visualization with other surfaces as many times as they like so that they will eventually gain intuitive understanding of geometrical concepts. The mathematics laboratory session continues like this.
5 CONCLUSIONS

Instructional patterns in mathematics classes at engineering departments are presented. These patterns help teachers to design courses, in which students play active roles performing various mathematical experiments. To demonstrate the effectiveness of the patterns, the topics in classical differential geometry are treated with an advanced computer algebra technology provided by Mathematica. Our toolkit is a collection of utility functions and patterns of combining such utilities.

Mathematical topics chosen to demonstrate the patterns are:
- covariant derivatives,
- geodesics, and
- parallel transport of tangent vectors.

Focused technical elements specific to Mathematica are:
- numerical solutions of differential equations in spline functions,
- dynamical objects in 3D graphics, and
- realistic way of producing smooth interactive 3D graphics with massive computation.

Future or on-going work includes
- Visual Levi-Civita Connection,
- Moving Frame Techniques,
- Intrinsic differential geometry, and
- More on Topology of Closed Surfaces.

The author hopes that these kind of practices are shared among teachers and students in various engineering fields.

REFERENCES


