Variational Methods for Discrete Surface Parameterization. Applications and Implementation.

vorgelegt von
Dipl.-Math. techn. Stefan Sechelmann

von der Fakultät II - Mathematik und Naturwissenschaften
der Technischen Universität Berlin
zur Erlangung des akademischen Grades

Doktor der Naturwissenschaften
– Dr. rer. nat. –

Promotionsausschuss

Vorsitzender:	NN
Gutachter/Berichter:	Prof. Dr. Alexander I. Bobenko
	NN

Tag der wissenschaftlichen Aussprache: NN

Berlin, den 07. Januar 2013

Contents

List of Figures

Chapter 1

Discrete Differential Geometry - Software Packages

1.1 Introduction

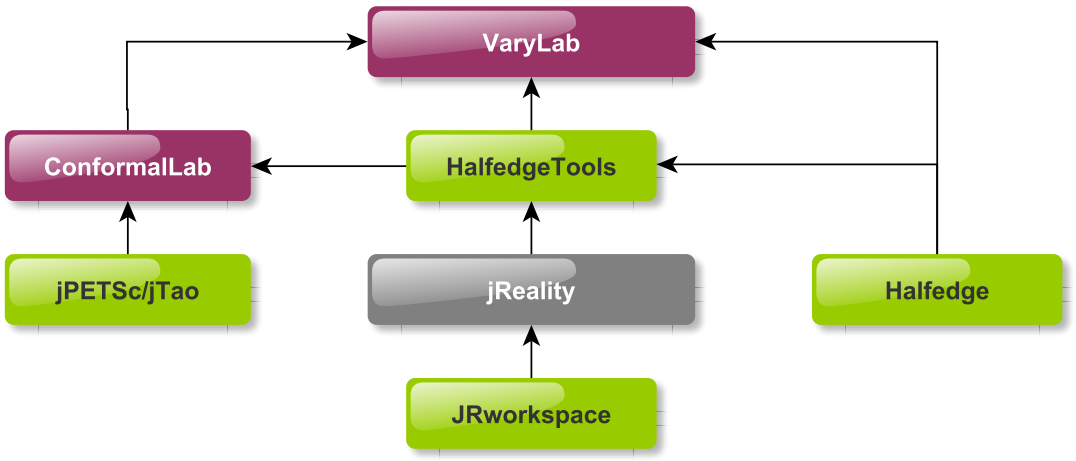

Figure 1.1: Software architecture and dependencies of the DDG Framework. JTEM library packages (green), mathematical software packages (red).

In the field of Discrete Differential Geometry (DDG) there is a special need for experiments conducted with the help of computer software. Especially if the methods of DDG are applied to problems in computer graphics, geometry processing, or architecture, algorithms have to be implemented and convincing examples have to be presented. Additionally a suitable visualization of the results has to be included in a state-of-the-art publication.

There is a growing knowledge of software development in the mathematical community. This is partly due to the curricula of universities which started to include programming courses for undergraduate students. **[Find Reference ??]** This enables the students to extend their abilities

`restoreState`	1.	load plug-in state values from Controller
`install`	2.	calls getPlugin to obtain other plug-ins
-	3.	program execution
`storeStates`	4.	stores state values in the Controller
`uninstall`/program terination	5.	clean up

Table 1.1: JRworkspace plug-in life cycle

of creating visualizations and mathematical software, where former generations of students solely used the visualization abilities of standard computer algebra packages like Mathematica or MatLab.

This Chapter is the description and getting-started manual of a set of software packages (Berlin DDG Framework) written in Java. They are specifically designed for the creation of custom interactive software for experiments with algorithms and geometries treated within DDG. Section 1.2 introduces the JRworkspace library of the Jtem project [jdt13b]. It is the foundation of any application created with the DDG Framework. It is also the user interface basis of Jreality, a mathematical visualization library that uses JRworkspace as plug-in and user interface tool [jdt13a]. Section 1.3 introduces the Halfedge and HalfedgeTools package. It implements a half-edge data structure and various user interface tools and algorithms for interaction and editing. In Section 1.4 we describe the software ConformalLab. This package implements the methods of the publications [BPS10, Sec12, SRB12, BSS]. Section 1.5 introduces VaryLab the software implementation of the methods described in the publications [LGSR11, LSRG12, SRB12]. This package is also released to partners of the development group as VaryLab[Gridshells], VaryLab[Ultimate], or even online as VaryLab[Service].

Figure 1.1 shows the dependencies of the packages. Every application depends on JRworkspace which implements plug-in functionality. It is the basis of the Jreality plug-in system. HalfedgeTools is using Jreality for visualization and is build on top of the Jtem project Halfedge. ConformalLab and VaryLab use Jpetsc/Jtao to perform numerical optimization. Their algorithms are implemented as JRworkspace plug-ins.

The development of the described software is joint work with Thilo Rörig (HalfedgeTools, VaryLab), the Jreality members [jdt13a], Hannes Sommer (Jpetsc/Jtao) [Som10], Ulrich Pinkall and Paul Peters (JRworkspace), and Boris Springborn (Halfedge).

1.2 JRworkspace - A Java API for modular applications

The library package JRworkspace is part of the Jtem family of software projects [jdt13b]. It defines a simple API to create modular Java applications. Figure 1.2 shows the UML class diagram of the three main classes. The project contains a reference implementation that supports the creation of Java Swing applications. This implementation is used in all applications described in this work.

1.2.1 Plug-ins

In JRworkspace a module is called plug-in and extends the abstract class Plugin (Figure 1.2). The idea is that a plug-in can be installed by the controller calling its `install` method or

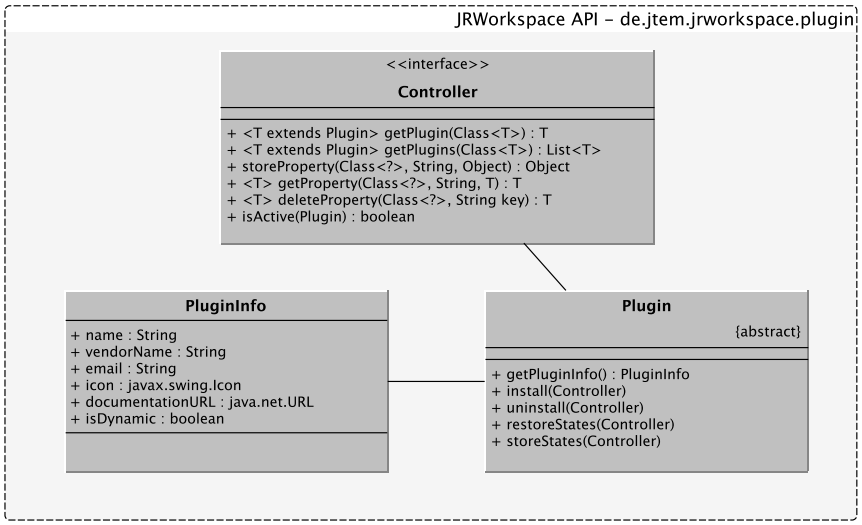

Figure 1.2: UML diagram for the JRWORKSPACE API.

uninstalled via the uninstall() method. Before installation of a plug-in the controller calles the `restoreStates` method of a plug-in. The `storeStates` method is called at program termination or right before uninstall of the plug-in. Inter-plug-in-communication is done via the `getPlugin` method of the controller. A plug-in should call `getPlugin` from within the install method to obtain an instance of a dependent plug-in. See Table 1.1 for the plug-in life-cycle.

```
1  public class MyPlugin extends Plugin {
2    private DependentPlugin dependency = null;
3    private double doubleState = 0.0;
4
5    public void helloPlugin() {
6      String depName = dependency.getPluginInfo().name;
7      System.out.println("I am a plug-in. I depend on " + depName);
8    }
9    @Override
10   public void storeStates(Controller c) throws Exception {
11     c.storeProperty(MyPlugin.class, "doubleState", doubleState);
12   }
13   @Override
14   public void restoreStates(Controller c) throws Exception {
15     doubleState = c.getProperty(MyPlugin.class, "doubleState", 1.0);
16   }
17   @Override
18   public void install(Controller c) throws Exception {
19     dependency = c.getPlugin(DependentPlugin.class);
20   }
21 }
```

Listing 1.1: A simple plug-in class. It depends on a plug-in called `DependentPlugin` and has the property `doubleState`. It provides the method `doWork` that prints some message.

Bibliography

[ACSD+03] Pierre Alliez, David Cohen-Steiner, Olivier Devillers, Bruno Lévy, and Mathieu Desbrun. Anisotropic polygonal remeshing. *ACM Trans. Graph.*, 22(3):485–493, 2003.

[BBB+11] Satish Balay, Jed Brown, Kris Buschelman, William D. Gropp, Dinesh Kaushik, Matthew G. Knepley, Lois Curfman McInnes, Barry F. Smith, and Hong Zhang. PETSc Web page, 2011. `http://www.mcs.anl.gov/petsc`.

[BHS06] Alexander I. Bobenko, Tim Hoffmann, and Boris Springborn. Minimal surfaces from circle patterns: geometry from combinatorics. *Ann. of Math. (2)*, 164(1):231–264, 2006.

[BMM+07] Steve Benson, Lois Curfman McInnes, Jorge Moré, Todd Munson, and Jason Sarich. TAO user manual (revision 1.9), 2007. `http://www.mcs.anl.gov/tao`.

[BP99] Alexander I. Bobenko and Ulrich Pinkall. Discretization of surfaces and integrable systems. In *Discrete integrable geometry and physics (Vienna, 1996)*, volume 16 of *Oxford Lecture Ser. Math. Appl.*, pages 3–58. Oxford Univ. Press, New York, 1999.

[BPS10] Alexander I. Bobenko, Ulrich Pinkall, and Boris Springborn. Discrete conformal maps and ideal hyperbolic polyhedra. Preprint; `http://arxiv.org/abs/1005.2698`, 2010.

[BS08] Alexander I. Bobenko and Yuri B. Suris. *Discrete differential geometry – Integrable structure*, volume 98 of *Graduate Studies in Mathematics*. American Mathematical Society, Providence, RI, 2008.

[BSS] Alexander I. Bobenko, Stefan Sechelmann, and Boris Springborn. Uniformization of discrete Riemann surfaces. in preparation.

[CSM03] David Cohen-Steiner and Jean-Marie Morvan. Restricted Delaunay triangulations and normal cycle. In *Symposium on Computational Geometry*, pages 312–321, 2003.

[FH05] M. S. Floater and K. Hormann. Surface parameterization: a tutorial and survey. In N. A. Dodgson, M. S. Floater, and M. A. Sabin, editors, *Advances in Multiresolution for Geometric Modelling*, Mathematics and Visualization, pages 157–186. Springer, Berlin, Heidelberg, 2005.

[Guo11] Ren Guo. Combinatorial yamabe flow on hyperbolic surfaces with boundary. *Commun. Contemp. Math.*, 13(5):827–842, 2011.

[HJ03] Udo Hertrich-Jeromin. *Introduction to Möbius Differential Geometry*. London Mathematical Society Lecture Note Series. Cambridge University Press, 2003.

[jdt13a] jReality developer team. jReality website, 2013. `http://www.jreality.de`.

[jdt13b] jTEM developer team. jTEM website, 2013. `http://www.jtem.de`.

[Jos07] Jürgen Jost. *Compact Riemann Surfaces*. Universitext (En ligne). Springer-Verlag Berlin Heidelberg, 2007.

[KNP07] Felix Kälberer, Matthias Nieser, and Konrad Polthier. Quadcover - surface parameterization using branched coverings. *Comput. Graph. Forum*, 26(3):375–384, 2007.

[LGSR11] Elisa Lafuente Hernández, Christoph Gengnagel, Stefan Sechelmann, and Thilo Rörig. On the materiality and structural behaviour of highly-elastic gridshell structures. In C. Gengnagel, A. Kilian, N. Palz, and F. Scheurer, editors, *Computational Design Modeling: Proceedings of the Design Modeling Symposium Berlin 2011*, pages 123–135. Springer, 2011.

[LPW+06] Yang Liu, Helmut Pottmann, Johannes Wallner, Yong-Liang Yang, and Wenping Wang. Geometric modeling with conical meshes and developable surfaces. *ACM Trans. Graph.*, 25(3):681–689, 2006.

[LSRG12] Elisa Lafuente Hernández, Stefan Sechelmann, Thilo Rörig, and Christoph Gengnagel. Topology optimisation of regular and irregular elastic gridshells by means of a non-linear variational method. In L. Hesselgren, S. Sharma, J. Wallner, N. Baldassini, P. Bompas, and J. Raynaud, editors, *Advances in Architectural Geometry 2012*, pages 147–160. Springer, 2012.

[Luo04] Feng Luo. Combinatorial yamabe flow on surfaces. *Commun. Contemp. Math.*, 6(5):765–780, 2004.

[PLW+07] Helmut Pottmann, Yang Liu, Johannes Wallner, Alexander Bobenko, and Wenping Wang. Geometry of multi-layer freeform structures for architecture. *ACM Trans. Graph.*, 26(3), July 2007.

[PSB+08] Helmut Pottmann, Alexander Schiftner, Pengbo Bo, Heinz Schmiedhofer, Wenping Wang, Niccolo Baldassini, and Johannes Wallner. Freeform surfaces from single curved panels. *ACM Trans. Graph.*, 27(3):#76, 1–10, 2008.

[SdS01] A. Sheffer and E. de Sturler. Parameterization of Faceted Surfaces for Meshing using Angle-Based Flattening. *Engineering with Computers*, 17:326–337, 2001.

[Sec12] Stefan Sechelmann. Uniformization of discrete Riemann surfaces. In *Oberwolfach Reports*, 2012.

[SHWP09] Alexander Schiftner, Mathias Höbinger, Johannes Wallner, and Helmut Pottmann. Packing circles and spheres on surfaces. *ACM Trans. Graphics*, 28(5):#139,1–8, 2009. Proc. SIGGRAPH Asia.

[Som10] Hannes Sommer. jPETScTao JNI library web page, 2010. `http://jpetsctao.zwoggel.net/`.

[SPR06] Alla Sheffer, Emil Praun, and Kenneth Rose. Mesh parameterization methods and their applications. *Foundations and Trends in Computer Graphics and Vision*, 2(2):105–171, 2006.

[Spr09] Boris Springborn. Discrete conformal equivalence for triangle meshes. In *Oberwolfach Reports*, volume 6 of *Oberwolfach Reports*, pages 104–106, 2009.

[SRB12] Stefan Sechelmann, Thilo Rörig, and Alexander I. Bobenko. Quasiisothermic mesh layout. In L. Hesselgren, S. Sharma, J. Wallner, N. Baldassini, P. Bompas, and J. Raynaud, editors, *Advances in Architectural Geometry 2012*, pages 243–258. Springer, 2012.

[SSP08] Boris Springborn, Peter Schröder, and Ulrich Pinkall. Conformal equivalence of triangle meshes. *ACM Trans. Graph.*, 27(3):77:1–77:11, August 2008.

[ZSW10] Mirko Zadravec, Alexander Schiftner, and Johannes Wallner. Designing quad-dominant meshes with planar faces. *Computer Graphics Forum*, 29(5):1671–1679, 2010. Proc. Symp. Geometry Processing.

Acknowledgements