

# Smoothed Quadratic Energies on Meshes

JANICK MARTINEZ ESTURO

Max Planck Institute for Informatics, Saarbrücken

CHRISTIAN RÖSSL and HOLGER THEISEL

Otto-von-Guericke University Magdeburg

In this paper, we study the regularization of quadratic energies that are integrated over discrete domains. This is a fairly general setting, which is often found in but not limited to geometry processing. The standard Tikhonov regularization is widely used such that, e.g., a low-pass filter enforces smoothness of the solution. This approach, however, is independent of the energy and the concrete problem, which leads to artifacts in various applications. Instead, we propose a regularization that enforces a low variation of the energy and is problem-specific by construction. Essentially, this approach corresponds to minimization w.r.t. a different norm. Our construction is generic and can be plugged into any quadratic energy minimization, is simple to implement, and has no significant runtime overhead. We demonstrate this for a number of typical problems and discuss the expected benefits.

Categories and Subject Descriptors: I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—*Geometric algorithms, languages, and systems*

General Terms: Algorithms

Additional Key Words and Phrases: Energy minimization, regularization, geometry processing

## ACM Reference Format:

Janick Martinez Esturo, Christian Rössl, and Holger Theisel. 2014. Smoothed Quadratic Energies on Meshes. ACM Trans. Graph. XX, XX, Article XX (XX XXX), 12 pages.

DOI: <http://dx.doi.org/XX>

## 1. INTRODUCTION

Solutions of a major class of problems in computer graphics are modeled by an energy minimization. Of particular interest are quadratic energies as their minimization leads to solving a linear system of equations — a task that is well-understood and relatively inexpensive.

---

© ACM, (2014). This is the author's version of the work. It is posted here by permission of ACM for your personal use. Not for redistribution. The definitive version was published in ACM Trans. Graph., XX, XX, (2014) <http://doi.acm.org/10.1145/XX>.

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies show this notice on the first page or initial screen of a display along with the full citation. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, to republish, to post on servers, to redistribute to lists, or to use any component of this work in other works requires prior specific permission and/or a fee. Permissions may be requested from Publications Dept., ACM, Inc., 2 Penn Plaza, Suite 701, New York, NY 10121-0701 USA, fax +1 (212) 869-0481, or [permissions@acm.org](mailto:permissions@acm.org).

© XX ACM 0730-0301/XX/14-ARTXX \$15.00

DOI: <http://dx.doi.org/XX>

For this reason, other types of energies that model more expensive nonlinear problems are often minimized iteratively such that every step is a minimization of a quadratic energy, e.g., in Quasi-Newton methods. Energy is a scalar quantity that is typically integrated over a domain. In discrete settings the domain is commonly partitioned into elements: energy density is defined locally on each element such that the global energy value consists of the sum of integrals over all elements. The simplest and most widespread partitions in computer graphics employ triangular elements for planar or two-manifold domains and tetrahedral elements for volumetric domains. In computer graphics, this setting is probably known best for geometry processing applications.

A lot of research has been devoted to the definition of energies and to their discretization (see, e.g., [Botsch et al. 2010]). In particular, this includes the well-known harmonic and conformal energies and discrete versions of gradient, divergence, and Laplace operators (or their counterparts in discrete exterior calculus). In this paper, we study the construction of associated *regularization terms* — a topic that has received considerably less attention in many applications. Regularization terms are generally used to enable the solution of ill-posed problems. An additional goal is to “pull” the solution towards a certain class of feasible solutions by penalizing “undesired” behavior, e.g., to prevent overfitting. A well-known and generic approach is Tikhonov regularization (see, e.g., [Hansen 2010]), which gives preference, for instance, to smoother solutions. De facto, this is the standard approach for regularizing the minimization of quadratic energies, especially for sparse operators.

In this paper, we propose a novel view on regularization that leads to a simple and generic construction of *problem-specific* regularized quadratic energies on triangular or tetrahedral partitions: instead of demanding smoothness of the solution, the key idea is to give preference to solutions with low spatial variation in the *energy* over the domain. This leads to the design of a regularization that is tailored to the particular energy and is problem-specific by construction. In other words, our proposed regularization refers to minimizing *smoothed* quadratic energies. This is in contrast to generic Tikhonov regularization methods, which are independent of the particular energy and hence “problem-independent”. We show that there is a *generic construction* for our regularization that is formally expressed as a generalized norm. Our approach features a simple and generic implementation and has no significant runtime overhead. We illustrate the recipe for the construction by applying our method to a number of typical problems in geometry processing (although it is not limited to problems of this field). The effect is demonstrated by experimental results that reveal the practical benefits in various applications, e.g., the avoidance of artifacts or in some instances the convergence to solutions that so far were only achieved by expensive nonlinear optimization.

## 2. BACKGROUND

Regularization is an established and important tool for solving numerical problems. Typical classes of application are the solution of rank-deficient problems or discrete ill-posed problems that frequently arise for inverse problems. We refer the reader to, e.g., [Hansen 2010] for an introduction and overview on regularization for this kind of problems. In general, the singular value decomposition reveals characteristics of linear operators, and regularization can be expressed as filtering of singular values, e.g., truncation in the simplest case. Classic Tikhonov regularization can be easily expressed as an explicit “filter” for singular values. In practice, this is often not an option as the dimension of linear operators may become too large. In particular, geometry processing problems involve too many degrees of freedom. Still, they typically employ sparse linear operators that define an integration of a quadratic energy in a domain. Therefore, an alternative Tikhonov regularization formulation is preferred in practice: adding a quadratic regularization energy term for implicitly filtering singular values. This way the numerical solution is typically obtained more efficiently from the normal equations or a QR-factorization.

There is a huge body of research on the appropriate discretization of linear operators. This becomes evident especially in the geometry processing context, where the domain is often a 2-manifold embedded in 3D space: the discretization of the Laplace-Beltrami operator by the well-known cotangent-weights [Pinkall and Polthier 1993] was only recently complemented by a formulation for general polygonal meshes [Alexa and Wardetzky 2011], and there are a number of variants (see, e.g., [Wardetzky et al. 2007]).

Interestingly, regularization operators received less attention. In fact, often regularization is not considered at all. Mostly there is a good reason for this: many problems are *not* ill-posed and don’t require regularization in order to be solvable. Moreover, energies are carefully designed with particular interpretations in mind, so from a puristic point of view regularization or additional penalty terms may spoil solutions. Probably, the most widely used “regularization” is the preference of “soft constraints” in the least-squares sense, i.e., introduction of a penalty term instead of elimination (see, e.g., [Botsch and Sorkine 2008]). If regularization takes place, then this is typically done by preferring either a low norm solution or a smooth solution. Both choices refer to standard Tikhonov regularization and do not take the particular choice of energy into consideration. A typical example is the reconstruction of curves and surfaces, where a smooth solution is preferred. Here, the additional smoothness term accounts for missing or inappropriately distributed data, that may render the problem ill-posed. For instance, this is the case in parametric spline fitting (see, e.g., [Hoschek and Lasser 1993]) or in more complex discrete settings (e.g., [Alliez et al. 2007]). Problem-specific energy regularization is only rarely applied in geometry processing: Eckstein et al. [2007] use regularization for the specific problem of curvature flow design. They construct specific regularizers or priors and show that this corresponds to defining new inner products or norm alteration. We will get to a similar interpretation for our generic approach without the restriction to geometric flow design.

Examples for ill-posed problems include the reconstruction of (geometric) data by Bayesian statistics, where the so-called prior distribution provides a suitable model of the data. The surface reconstruction in [Huang et al. 2007], for instance, uses the prior to constrain a “prototype surface”. Generally, the prior restricts the solution space and enables a meaningful solution; this can be seen as a regularization as well. Another notable example of a problem that is likely to be ill-posed is the estimation of a linear operator for the

refinement of animations: Kavan et al. [2011] transform a quadratic energy into a spectral space where the regularization operator, a spatial low-pass filter, becomes diagonal.

The applications shown in this paper focus on typical geometry processing tasks. In contrast to the examples listed above, regularization is usually an optional part in this domain as problems are generally well-posed (subject to boundary constraints). We show that using regularization is worthwhile, if it is done carefully in a problem-specific way. Although our approach is general and not restricted to the geometry processing domain, we mostly stick to this perspective in the following. We selected a number of standard problems and representative state-of-the-art solutions as examples for employing our approach. Credits to these original works as well as brief summaries are given in the respective parts of Section 5.

## 3. SMOOTHED ENERGIES

For our generic regularization we start with the study of a general class of optimization problems. We provide concrete examples for such problems later in Section 5. For a compact spatial domain  $\mathcal{D}$ , a general optimization problem  $\mathcal{P}$  consists of finding the  $d$ -dimensional function  $\mathbf{u} \in \mathcal{F}$ ,  $\mathbf{u}: \mathcal{D} \rightarrow \mathbb{R}^d$  from the function space  $\mathcal{F}$  of feasible solutions<sup>1</sup> that minimizes a problem-specific global quadratic energy  $E_{\mathcal{P}}$ . These *global energies* (or energy functionals) have the general form<sup>2</sup>

$$E_{\mathcal{P}}(\mathbf{u}) = \int_{\mathcal{D}} \|\mathbf{e}_{\mathbf{u}}(\mathbf{x})\|^2 d\mathbf{x}. \quad (1)$$

Here,  $\mathbf{x} \in \mathcal{D}$  are points in the domain where *local energies* (or, more precisely, energy densities)  $\|\mathbf{e}_{\mathbf{u}}\|^2 : \mathcal{D} \rightarrow \mathbb{R}$  of the global energies are measured. Local energies are defined by vector-valued differentiable and square-integrable *local energy components*  $\mathbf{e}_{\mathbf{u}}: \mathcal{D} \rightarrow \mathbb{R}^n$ . Local energy components evaluate the problem-specific quality of a feasible solution  $\mathbf{u}$  of  $\mathcal{P}$  at each point of the domain. Their dimension  $n$  depends on the concrete problem  $\mathcal{P}$ , and  $\mathbf{e}_{\mathbf{u}}$  is *linear* in  $\mathbf{u}$ , i.e.,  $E_{\mathcal{P}}$  is quadratic in  $\mathbf{u}$ . The spatial domains may be manifolds, e.g., 2-manifolds embedded in 3D. An optimal solution

$$\mathbf{u}^\dagger = \operatorname{argmin}_{\mathbf{u} \in \mathcal{F}} E_{\mathcal{P}}(\mathbf{u}) \quad (2)$$

is a minimizer of  $E_{\mathcal{P}}$ , which is obtained as the solution of a linear system. This optimization generally has to be performed subject to suitable boundary constraints on  $\mathbf{u}^\dagger$ . Boundary constraints may be required to guarantee unique solutions, and/or they are imposed by the user to modify the solution.

Many geometry processing problems  $\mathcal{P}$  are in this class because they require minimization of quadratic energies. For example,  $\mathcal{D}$  could represent an initial surface, and  $\mathbf{u}$  would be the coordinate function of a deformed version of  $\mathcal{D}$ . In this case,  $\mathbf{e}_{\mathbf{u}}$  measures some form of distortion induced by the deformation  $\mathbf{u}$  relative to the initial shape given by  $\mathbf{x}$ . Usually,  $\mathbf{e}_{\mathbf{u}}$  only depends on the value and the gradient of  $\mathbf{u}$ . We will show several examples in Section 5.

Often energies of the form (1) alone are not sufficient to define *desirable* solutions, as the local energy components  $\mathbf{e}_{\mathbf{u}}$  do not measure smoothness of the solution. This leads to undesirable artifacts in  $\mathbf{u}^\dagger$ , which are often most noticeable near user-imposed constraints. To alleviate undesired behavior, a common approach is to add an

<sup>1</sup>We assume that all derivatives of  $\mathbf{u}$  required for  $\mathcal{P}$  exist and are bounded by the selection of  $\mathcal{F}$ , e.g.,  $\mathcal{F} \equiv H^1(\mathcal{D}, \mathbb{R}^d)$ .

<sup>2</sup>We denote a domain-wide integration of scalar densities  $s(\mathbf{x})$  along the volume elements  $d\mathbf{x}$  by  $\int_{\mathcal{D}} s(\mathbf{x}) d\mathbf{x}$  (see, e.g., [Eckstein et al. 2007]).

additional quadratic regularization energy  $E_{\mathcal{R}}(\mathbf{u})$  that penalizes non-smooth or in other respects undesirable behavior of  $\mathbf{u}^\dagger$ :

$$E(\mathbf{u}) = (1 - \beta) E_{\mathcal{P}}(\mathbf{u}) + \beta E_{\mathcal{R}}(\mathbf{u}) \quad (3)$$

$$\mathbf{u}^* = \operatorname{argmin}_{\mathbf{u} \in \mathcal{F}} E(\mathbf{u}). \quad (4)$$

Here,  $\mathbf{u}^*$  denotes the optimal regularized solution of  $\mathcal{P}$  that minimizes the regularized energy  $E$ . The amount of regularization is steered by the factor  $\beta \in [0, 1]$ . These energies often use forms of generic *Tikhonov regularizers*  $E_{\mathcal{R}}^T$  for  $E_{\mathcal{R}}$  that have the general form (see, e.g., [Hansen 2010] and the references therein):

$$E_{\mathcal{R}}^T(\mathbf{u}) = \int_{\mathcal{D}} \|\Gamma_{\mathbf{u}}(\mathbf{x})\|_F^2 d\mathbf{x}. \quad (5)$$

Here,  $\|\cdot\|_F$  denotes the Frobenius norm, and a prominent choice for the linear regularization operator  $\Gamma_{\mathbf{u}}$  is given by the componentwise spatial gradient operator  $\Gamma_{\mathbf{u}} \equiv \nabla \mathbf{u}$  on  $\mathcal{D}$  to penalize first-order variation of the solution and to enforce global smoothness of  $\mathbf{u}$ . Other common examples include  $\Gamma_{\mathbf{u}} \equiv \mathbf{I} \mathbf{u}$  or  $\Gamma_{\mathbf{u}} \equiv \nabla^2 \mathbf{u}$  that penalize solution magnitude or second-order variation, respectively. Similar types of generic regularizations are also well-known in other application domains, e.g., in image processing [Strong and Chan 2000], where they are called *total variation regularization*.

We identify two major drawbacks of using these standard regularizations: first, generic regularizers like those based on a low-pass filter on  $\mathbf{u}$  are *independent* of the concrete problem that is defined by the energy  $E_{\mathcal{P}}$ . We call this property *problem-independence*. They often perform poorly or too aggressively in correcting artifacts in  $\mathbf{u}^\dagger$  (see Section 7). In fact, standard first-order Tikhonov regularization modifies solutions even if  $E_{\mathcal{P}}$  already vanishes. Second, the design of effective regularizations is a delicate problem, and it is often non-obvious as the original properties of  $E_{\mathcal{P}}$  should not be affected too much.

**Energy Smoothness.** To overcome these drawbacks, we propose a new regularization that is tightly coupled to the minimized energy and is based on *energy smoothness*. We define *energy smoothness regularization* as

$$E_{\mathcal{R}}(\mathbf{u}) = \int_{\mathcal{D}} \|\nabla \mathbf{e}_{\mathbf{u}}(\mathbf{x})\|_F^2 d\mathbf{x}, \quad (6)$$

i.e., as the total squared spatial variation of the local energy components. In contrast to functionals like (5) we do not directly measure variation of the solution  $\mathbf{u}$ , but variation of the induced local energy components  $\mathbf{e}_{\mathbf{u}}$ . This way the regularization is tightly coupled and tailored to the original energy (1) and is *problem-specific*. With respect to this property  $E_{\mathcal{R}}$  and the common Tikhonov regularizers  $E_{\mathcal{R}}^T$  are generally not identical, which will be discussed in Section 7.

Our regularization is motivated by the observation that artifacts usually result in strong local energy variations. These variations are effectively penalized by our approach. This results in solutions for which energy is distributed more evenly in the whole domain, which indirectly prevents artifacts: we demonstrate in Section 6 that for a wide range of problems attractive results are obtained by this conceptually simple but effective regularization strategy. Due to the linearity of the gradient operator, the tensor field  $\nabla \mathbf{e}_{\mathbf{u}}$  is also linear in  $\mathbf{u}$ . Hence, the regularized optimization (4) stays quadratic in  $\mathbf{u}$  and is efficient to compute. Moreover, once a discretization of the original problem (2) is set up, the regularization using (6) is very simple to obtain. We show this in the next section.

## 4. DISCRETIZATION

In the following we consider only two-dimensional domains as the construction is essentially the same for three dimensions. We discretize planar or two-manifold domains  $\mathcal{D}$  by triangular meshes  $\mathcal{M} = (\mathcal{V}, \mathcal{E}, \mathcal{T})$ , i.e., as sets of vertices  $i \in \mathcal{V}$ , oriented edges  $\mathcal{E} \subset \mathcal{V}^2$ , and triangles  $\mathcal{T} \subset \mathcal{V}^3$ . The edges  $\mathcal{E}$  can be partitioned into two disjoint sets: interior edges  $\mathcal{E}_i$  and exterior edges  $\mathcal{E} \setminus \mathcal{E}_i$ , the latter positioned at the boundary of  $\mathcal{M}$ . We write  $l(e)$  and  $r(e)$  to denote the left and right triangle at an oriented internal edge  $e$ , respectively. We consider discrete functions on  $\mathcal{M}$  that are piecewise linear and represented by coefficients at the vertices, e.g., the vertex coordinate function  $\mathbf{x}_i \in \mathbb{R}^{2/3}$ , or problem solutions  $\mathbf{u}_i \in \mathbb{R}^d$ . All coefficients of a function are “stacked” into a single vector without subscript, e.g.,  $\mathbf{u} \in \mathbb{R}^{d|\mathcal{V}|}$ , or stacked componentwise into a single matrix  $\mathbf{U} \in \mathbb{R}^{|\mathcal{V}| \times d}$ . For a triangle  $t = (i, j, k) \in \mathcal{T}$  we denote the stacked coefficients at all vertices of  $t$  by  $\mathbf{u}_t = (\mathbf{u}_i^T, \mathbf{u}_j^T, \mathbf{u}_k^T)^T \in \mathbb{R}^{3d}$ , and the component-wise stacked coefficient matrix by  $\mathbf{U}_t = (\mathbf{u}_i, \mathbf{u}_j, \mathbf{u}_k)^T \in \mathbb{R}^{3 \times d}$ . For convenience, we define the single-entry  $r \times s$  matrix  $\Lambda_{r,s}^{i,j}$  that is 1 at  $(i, j)$  and zero elsewhere. Finally, we use the notation

$$\|\mathbf{y}\|_{\mathbf{N}}^2 = \mathbf{y}^T \mathbf{N} \mathbf{y} \quad \text{and} \quad \|\mathbf{Y}\|_{\mathbf{N}}^2 = \operatorname{Tr}(\mathbf{Y}^T \mathbf{N} \mathbf{Y}) \quad (7)$$

for (squared) vector and matrix norms that are induced by symmetric and positive definite matrices  $\mathbf{N}$ . ( $\operatorname{Tr}()$  denotes the trace of a matrix.) For instance, using the identity matrix  $\mathbf{I}$  the well-known Frobenius norm  $\|\cdot\|_F$  is equivalent to  $\|\cdot\|_{\mathbf{I}}$  in this notation.

### 4.1 Piecewise Constant Energies

The majority of geometry processing approaches discretize continuous energies on a per triangle basis, e.g., Poisson-based methods (see, e.g., [Botsch et al. 2010]). In these discretizations, the local energy components  $\mathbf{e}_{\mathbf{u}}$  of  $E_{\mathcal{P}}$  are *constant* on each triangle, and domain integration is performed by area weighting. Most discretizations of global, purely quadratic piecewise constant energies can be expressed in the form

$$E_{\mathcal{P}}(\mathbf{u}) = \|\mathbf{E} \mathbf{u} - \mathbf{c}\|_{\mathbf{A}_n}^2. \quad (8)$$

Here,  $\mathbf{E}(\mathbf{x})$  is a global energy matrix of dimension  $n |\mathcal{T}| \times d |\mathcal{V}|$ , and  $\mathbf{c}(\mathbf{x}) \in \mathbb{R}^{n|\mathcal{T}|}$  is triangle-constant.  $\mathbf{E}$  maps feasible solutions  $\mathbf{u}$  to  $|\mathcal{T}|$  consecutive  $n$ -dimensional local energy vectors  $\mathbf{e}_{\mathbf{u},t}$ , which are constant per triangle  $t$ . In most cases,  $\mathbf{E}$  is a sparse matrix. In the remainder of this work, we assume that all  $\mathbf{e}_{\mathbf{u},t}$  are defined in a common coordinate system, e.g., the canonical world coordinates in  $\mathbb{R}^3$ . This can always be ensured by construction. Then  $\|\mathbf{e}_{\mathbf{u},t} - \mathbf{c}_t\|^2$  is also constant per triangle. Its integration over the mesh is performed by the  $n |\mathcal{T}|$ -dimensional square diagonal matrix  $\mathbf{A}_n(\mathbf{x})$  given by  $\mathbf{A}_n = \mathbf{A} \otimes \mathbf{I}_n$ . Here, we denote the Kronecker product by  $\otimes$ ,  $\mathbf{I}_n$  is the  $n \times n$  identity matrix, and  $\mathbf{A}(\mathbf{x})$  is the  $|\mathcal{T}| \times |\mathcal{T}|$  diagonal matrix of triangle areas.

For applications where the coordinate functions of the solutions are decoupled,  $\mathbf{E}$  can be expressed as a matrix of dimension  $n |\mathcal{T}| \times |\mathcal{V}|$ , and the vectors  $\mathbf{u}$  and  $\mathbf{c}$  become matrices  $\mathbf{U}$  and  $\mathbf{C}$  whose  $d$  columns correspond to coordinate functions. The general formulation (8) remains the same with the difference that it expresses a matrix norm (see (7)).

**Energy Smoothness.** Given a discretized energy in the form (8), the derivation of a discretization of our energy smoothness

regularization  $E_R$  of (6) turns out to be straightforward. This is because the gradient field  $\nabla \mathbf{e}_u$  vanishes almost everywhere on  $\mathcal{M}$  as  $\mathbf{e}_u$  is constant on each triangle:  $\mathbf{e}_u$  varies only across internal edges. Therefore, the tensor  $\nabla \mathbf{e}_u$  has a directional derivative component only orthogonal to each internal edge at each point of the edge. We estimate these directional derivatives by finite differences. A discretization of the energy smoothness regularization  $E_R$  that is directly based on the discrete energy  $E_P$  is then given by

$$E_R(\mathbf{u}) = \|\mathbf{D}_n (\mathbf{E} \mathbf{u} - \mathbf{c})\|_{\mathbf{B}_n}^2.$$

Here,  $\mathbf{D}_n$  is a sparse  $n |\mathcal{E}_i| \times n |\mathcal{T}|$  discrete differential operator on the local energy components. It is given by  $\mathbf{D}_n = \mathbf{D} \otimes \mathbf{I}_n$ , where the  $|\mathcal{E}_i| \times |\mathcal{T}|$  differential matrix  $\mathbf{D}$  is nonzero only at  $D_{et} = \begin{cases} 1 & \text{if } l(e) = t \\ -1 & \text{if } r(e) = t \end{cases}$ , for all internal edges  $e \in \mathcal{E}_i$  and triangles  $t \in \mathcal{T}$ . As  $\mathbf{e}_u$  is constant along each side of an edge, the integration of (6) simplifies to a weighted sum of quadratic directional differences of the local energy along each internal edge. This integration is performed by the  $n |\mathcal{E}_i|$  square diagonal matrix  $\mathbf{B}_n(\mathbf{x})$  given by  $\mathbf{B}_n = \mathbf{B} \otimes \mathbf{I}_n$ , where  $\mathbf{B}(\mathbf{x})$  is the diagonal matrix of edge lengths of all internal edges. Ultimately, this discretization of (6) can be interpreted as measuring the squared differences of constant local energy components along each internal edge, which are integrated by weighting with the respective edge lengths. Due to this tessellation-dependent discretization, the regularization is insensitive to the particular tessellation of the domain (see Section 6). The edge length weights can further be modulated to obtain spatially varying regularizations.

A discretization of the regularized energy (3) is then given by

$$E(\mathbf{u}) = (1 - \beta) \|\mathbf{E} \mathbf{u} - \mathbf{c}\|_{\mathbf{A}_n}^2 + \beta \|\mathbf{D}_n (\mathbf{E} \mathbf{u} - \mathbf{c})\|_{\mathbf{B}_n}^2.$$

The formulation of this energy allows for a transformation (see Appendix) into an equivalent expression

$$E(\mathbf{u}) = \|\mathbf{E} \mathbf{u} - \mathbf{c}\|_{\mathbf{W}_n}^2 \quad (9)$$

$$\text{with } \mathbf{W}_n = (1 - \beta) \mathbf{A}_n + \beta \mathbf{D}_n^T \mathbf{B}_n \mathbf{D}_n.$$

Our regularization has the following properties:

**$\beta$ -weighted Norm.** The total regularized energy in (9) has the remarkable property that it measures a different norm of the *same* energy residual  $\mathbf{E} \mathbf{u} - \mathbf{c}$  as (8). In fact, the domain integration represented by  $\mathbf{A}_n$  simply has to be replaced by the  $\beta$ -weighted *generalized norm* induced by  $\mathbf{W}_n(\mathbf{x}, \beta)$ . We show in the Appendix that the norm induced by  $\mathbf{W}_n$  is well-defined.

For  $\beta = 0$ , the original energy is recovered, and  $\mathbf{W}_n$  only performs energy integration. For  $\beta > 0$ ,  $\mathbf{W}_n$  also measures first-order energy variation. Instead of energy residuals any other triangle-constant function may be integrated using  $\mathbf{W}_n$ . Generally,  $\mathbf{W}_n$  induces a norm on the function space of triangle-constant functions. Minimization in this space yields functions of low norms, which incorporates low energy variation as steered by  $\beta$ . As the minimization problem is defined by the energy, penalizing energy variation results in a problem-specific regularization. In Section 6, we show for a number of typical applications that solutions  $\mathbf{u}^*$  of smoothed energies  $E$  defined by  $(\mathbf{E}, \mathbf{c})$  are preferable to solutions without regularization or to those from using Tikhonov regularization.

The norm induced by  $\mathbf{W}_n$  can be interpreted as a weighted Sobolev  $H^1$  norm, i.e., as a norm in the function value *and* its first-order variation. Sundaramoorthi et al. [2007] and Eckstein et al. [2007] use such a norm for the specific problem of computing smoothed gradient flows. In contrast, we provide a general con-

struction for smooth energy minimization that is not restricted to gradient flows, but supports any type of energies on meshes, e.g., parametrization energies. This is a key difference that enables the general application of our method to a variety of geometry processing problems.

**Generic Implementation.** The generalization of  $\|\cdot\|_{\mathbf{A}_n}$  to  $\|\cdot\|_{\mathbf{W}_n}$  is also very beneficial from an implementation point of view: many problems that employ quadratic energies on meshes are already represented in the discretized form (8). Given a model of the problem as  $(\mathbf{E}, \mathbf{c})$ , adding a problem-specific regularization is nothing more than replacing the integration  $\mathbf{A}_n$  by  $\mathbf{W}_n$ , i.e., to minimize (9). The setup of the matrix  $\mathbf{W}_n$  is simple, it depends only on the connectivity and coordinates of the domain mesh  $\mathcal{M}$ . In particular, the setup of  $\mathbf{W}_n$  is independent of a concrete energy or concrete problem, and can be reused. This makes our energy smoothness regularization by  $\mathbf{W}_n$  a generic function that takes as input  $\beta$  together with a mesh  $\mathcal{M}$  and can be applied to *any* problem  $(\mathbf{E}, \mathbf{c})$ .

**Optimization.** The minimizer  $\mathbf{u}^*$  of the discretization of (4) given by (9) can be computed by solving for the critical point of  $\nabla E(\mathbf{u})$ . Using normal equations, this is equivalent to solving the linear system

$$\mathbf{E}^T \mathbf{W}_n \mathbf{E} \mathbf{u}^* = \mathbf{E}^T \mathbf{W}_n \mathbf{c}. \quad (10)$$

After a potential elimination of boundary constraints, which might be required due to rank deficiencies of  $\mathbf{E}$ , the system becomes symmetric positive definite and can be solved with the Cholesky factorization  $\mathbf{L} \mathbf{L}^T = \mathbf{E}^T \mathbf{W}_n \mathbf{E}$  (see, e.g., [Botsch and Sorkine 2008]). Depending on the application, the factor can be reused for different right-hand sides  $\mathbf{c}$ . This factorization is efficient also for sparse matrices if an additional fill-in reducing reordering is applied. Obviously, if  $\mathbf{E}$  is sparse then the system matrix remains sparse. However, for  $\beta > 0$  the number of nonzero entries increases as  $\mathbf{W}_n$  is only “quasi-banded” but not diagonal anymore, whereas  $\mathbf{A}_n$  is. Our experiments indicate that this lower sparseness results in almost no loss in performance (see Section 6). This is because the dimension of the systems, which dominates the asymptotic runtime of linear solvers, is unchanged.

**Volumetric Domains.** The discretization is not restricted to triangulations of 2D domains. The setting is very similar for 3D domains. There, a tetrahedral partition of the domain is considered, and the local energies are constant functions defined per tetrahedron. Then the diagonal matrix  $\mathbf{A}_n$  of tetrahedra volumes performs the integration, and the differential operator  $\mathbf{D}_n$  is defined for inner triangles, each shared by its left and right tetrahedron. Consequently, the integration of squared energy variation by  $\mathbf{B}_n$  is then based on triangle areas instead of edge lengths.

## 4.2 Piecewise Linear Energies

Until now, we consider only local energies that are constant on each triangle. A similar approach is possible for *piecewise linear* local energies that are defined as nodal values at vertices. Note that it is generally easier and more natural to define energies or errors on a per-triangle basis. For example, all of our applications in Section 5 will only require piecewise constant energies. Still, for instance energies used in piecewise linear finite element methods (FEM) are often discretized on a per-vertex basis. Therefore, we give a short outline for the piecewise linear energy setting, for which our concept is applicable as well.

Quadratic piecewise linear energies on triangle meshes can again be written in the general form

$$E_{\mathcal{P}}(\mathbf{u}) = \|\mathbf{E} \mathbf{u} - \mathbf{c}\|_{\mathbf{M}_n}^2.$$

Here,  $\mathbf{E}(\mathbf{x})$  is the energy matrix of dimension  $n |\mathcal{V}| \times d |\mathcal{V}|$ , and  $\mathbf{c}(\mathbf{x}) \in \mathbb{R}^{n|\mathcal{V}|}$  is constant per vertex. In contrast to (8), energies are piecewise linear and we perform domain-wide energy integration by the  $n |\mathcal{V}| \times n |\mathcal{V}|$  FEM mass matrix  $\mathbf{M}_n(\mathbf{x})$  that integrates each of the  $n$  energy components separately (see, e.g., [Braess 2007]). For regularization, we measure the first-order energy variation by integrated energy gradients that are triangle-constant:

$$E_{\mathcal{R}}(\mathbf{u}) = \|\mathbf{G}_n (\mathbf{E} \mathbf{u} - \mathbf{c})\|_{\mathbf{A}_{mn}}^2.$$

Here,  $\mathbf{G}_n$  is the  $mn |\mathcal{T}| \times n |\mathcal{V}|$  gradient operator that assigns each triangle the constant gradient vector of each of the  $n$  energy components in a canonical  $m$ -dimensional basis, e.g.,  $m = 2$  for triangle meshes and  $m = 3$  for tetrahedral meshes.  $\mathbf{G}_n$  can be constructed in a component-wise way by replicating the plain two-variate gradient operator  $\mathbf{G}$  on  $\mathcal{M}$  (Equation (16) in the next section provides an example). Integration of the constant squared gradient norms is then again performed by area-based weighting using the  $mn |\mathcal{T}|$  diagonal triangle area matrix  $\mathbf{A}_{mn}$ .

The regularization of piecewise linear energies can again be expressed compactly as

$$E(\mathbf{u}) = \|\mathbf{E} \mathbf{u} - \mathbf{c}\|_{\mathbf{W}_n}^2$$

$$\text{with } \mathbf{W}_n = (1 - \beta) \mathbf{M}_n + \beta \mathbf{G}_n^T \mathbf{A}_{mn} \mathbf{G}_n.$$

Hence, similar to piecewise constant energies, we obtain regularized solutions by optimization in a space of smoother solutions expressed by the modified norm  $\mathbf{W}_n$ .

Note that different integration operators are commonly used in the literature, e.g., diagonal lumped barycentric mass matrices [Eckstein et al. 2007; Botsch and Sorkine 2008]. These operators can be incorporated into our framework in a similar way. However, depending on the particular integration model, different forms of energy gradient operators need to be discretized. For instance, for constant barycentric integration, energy gradients need to be estimated along the internal dual edges of the mesh.

## 5. APPLICATIONS

We demonstrate our approach for a number of typical problems  $\mathcal{P}$  from geometry processing. The minimization of quadratic energies on meshes is ubiquitous in this application domain. The particular problems, their discretizations and solutions have been addressed in a range of prior work. We reference representative work and describe how the problem-specific discretization ( $\mathbf{E}, \mathbf{c}$ ) is set up. We try to keep this description abstract to emphasize the main differences of the particular settings. We do not recap the setup and elimination of boundary conditions, which is discussed, e.g., in [Botsch and Sorkine 2008]. Most boundary conditions are “hard” interpolation constraints like, for instance, Dirichlet conditions. Adding our regularization is then a simple and generic procedure as described above. Our examples include linear 2D shape deformations, nonlinear deformations in 2D and 3D, planar and volumetric continuous deformations, as well as surface parametrization. Note that none of the problems is ill-posed in the first place and all problems are solvable even without regularization. We show, however, that all approaches are enhanced by our regularization, i.e., by minimizing smoothed quadratic energies.

### 5.1 Linear Planar Deformations

Shape deformations  $\mathcal{P}$  of two-dimensional planar meshes  $\mathbf{x}_i \in \mathbb{R}^2$  are displacement maps  $\mathbf{u}(\mathbf{x}) : \mathbb{R}^2 \rightarrow \mathbb{R}^2$  with  $\mathbf{u}(\mathbf{x}_i) = \mathbf{u}_i$ ,  $i \in \mathcal{V}$  that are piecewise linear on each triangle  $t \in \mathcal{T}$ :  $\mathbf{u}(\mathbf{x}) = \mathbf{F}_t \mathbf{x} + \mathbf{t}_t$ . A common approach to defining energy minimizing elastic deformations is to measure the difference of a particular deformation from a prototype deformation. Then the optimal deformation is given by the energy minimizing solution  $\mathbf{u}^*$ . As deformations are piecewise linear, the  $2 \times 2$  deformation gradient tensor field  $\mathbf{F}_t(\mathbf{u})$  is constant on each triangle. Isotropically integrated translation-invariant discrete deformation energies are therefore given by

$$E_{\mathcal{P}}(\mathbf{u}) = \sum_{t \in \mathcal{T}} A_t \|\mathbf{F}_t - \mathbf{T}(\mathbf{F}_t)\|_F^2, \quad (11)$$

with triangle areas  $A_t$  and closest  $2 \times 2$  prototype deformation gradient tensors  $\mathbf{T}(\mathbf{F}_t)$ . Different choices for  $\mathbf{T}$  will be discussed in the following paragraphs. As  $\mathbf{F}_t$  is the gradient of  $\mathbf{u}$  on a triangle  $t$ , it can be computed using a  $2 \times 3$  gradient operator  $\mathbf{G}_t(\mathbf{x})$ :  $\mathbf{F}_t = \mathbf{G}_t \mathbf{U}_t$ . We use a 2D gradient operator that computes gradients in a common global coordinate system to be able to apply our regularization. It is given for a triangle  $t = (i, j, k)$  by

$$\mathbf{G}_t = \begin{bmatrix} (\mathbf{x}_j - \mathbf{x}_i)^T \\ (\mathbf{x}_k - \mathbf{x}_i)^T \end{bmatrix}^{-1} \begin{pmatrix} -1 & 1 & 0 \\ -1 & 0 & 1 \end{pmatrix} \quad (12)$$

(see, e.g., [Botsch and Sorkine 2008] for a derivation). It is convenient to define a linear  $4 \times 6$  operator  $\mathbf{H}_t = [\mathbf{G}_t^T \ \mathbf{G}_t]$   $\mathbf{P}$  with an appropriate permutation  $\mathbf{P}$ , such that  $\text{vec}(\mathbf{F}_t) = \mathbf{H}_t \mathbf{u}_t$  is the column-wise vectorization of  $\mathbf{F}_t$ , i.e., the stacked gradients of all displacement coordinate functions.

*As-similar-as-possible (ASAP).* We call deformations as-similar-as-possible (ASAP) if they are approximately conformal, i.e., angles should be preserved. Here, we use a characterization that is similar to the one given by Liu et al. [2008]. Discrete 2D deformations are conformal in the least-squares sense if all deformation gradients  $\mathbf{F}_t = \begin{pmatrix} f_{11} & f_{12} \\ f_{21} & f_{22} \end{pmatrix}$  are as close as possible to the closest 2D similarity matrix  $\mathbf{S} \in \left\{ \begin{pmatrix} a & b \\ -b & a \end{pmatrix} : a, b \in \mathbb{R} \right\}$  that minimizes  $\|\mathbf{F}_t - \mathbf{S}\|_F^2$ . The similarity minimizing this distance is

$$\mathbf{S}_t^* = \frac{1}{2} \begin{pmatrix} f_{11} + f_{22} & f_{12} - f_{21} \\ f_{21} - f_{12} & f_{11} + f_{22} \end{pmatrix}.$$

The prototype deformation tensor in (11) is therefore given by the closest similarity  $\mathbf{T}(\mathbf{F}_t) = \mathbf{S}_t^*$ .  $\mathbf{T}$  is linear in  $\mathbf{F}_t$  for this case of angle-preserving or as-similar-as-possible deformations. Each triangle-constant local energy term of the summation in (11) can thus be written as  $\|\mathbf{E}_t \mathbf{u}_t\|^2$  with local  $4 \times 6$  energy operators

$$\mathbf{E}_t = \frac{1}{2} \begin{pmatrix} 1 & 0 & 0 & -1 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ -1 & 0 & 0 & 1 \end{pmatrix} \mathbf{H}_t.$$

The local operators  $\mathbf{E}_t$  on individual triangles can then be assembled into the single global  $4 |\mathcal{T}| \times 2 |\mathcal{V}|$  operator  $\mathbf{E}$  to give the energy (11) in the general form (8):

$$\mathbf{E} = \left( \sum_{t \in \mathcal{T}} \Lambda_{|\mathcal{T}|, |\mathcal{T}|}^{t,t} \otimes \mathbf{E}_t \right) \mathbf{Q}, \quad \mathbf{c} = \mathbf{0}. \quad (13)$$

Here,  $\mathbf{Q}$  is an appropriate replication matrix that selects all  $\mathbf{u}_i$  associated with a particular triangle, and  $\Lambda_{|\mathcal{T}|, |\mathcal{T}|}^{t,t}$  is a single-entry matrix (see Section 4). This description of the as-similar-as-possible

deformation energy is then suitable for regularization by (9). For this problem, we have  $n = 4$  as each triangle has four associated energy components.

*Linearized as-rigid-as-possible (LARAP).* 2D deformations that are as-rigid-as-possible are commonly computed by optimizing for deformation gradients that are close to rotations. Due to the non-linearity of rotations, this optimization generally requires iterative schemes (see Section 5.2). Linearization of rotations is used to avoid nonlinear problems. However, this simplification is often considered defective as large rotations lead to linearization artifacts. Still, we show that even this linearized setting gives more competitive results with our smoothed energy.

The rotation  $\mathbf{R} \in \left\{ \begin{pmatrix} \cos(\alpha) & \sin(\alpha) \\ -\sin(\alpha) & \cos(\alpha) \end{pmatrix} : \alpha \in \mathbb{R} \right\}$  that minimizes  $\|\mathbf{F}_t - \mathbf{R}\|_F^2$  has an optimal rotation angle of  $\alpha^* = \tan^{-1}(a)$  with  $a = f_{12} - f_{21}/f_{11} - f_{22}$ , and is given by

$$\mathbf{R}_t^* = \frac{1}{\sqrt{a^2 + 1}} \begin{pmatrix} 1 & a \\ -a & 1 \end{pmatrix}. \quad (14)$$

A linearized approximation  $\mathbf{R}_t^\dagger \approx \mathbf{R}_t^*$  is obtained from a Taylor series expansion of  $\mathbf{R}_t^*$  around  $\mathbf{I}_2$ :

$$\mathbf{R}_t^\dagger = \frac{1}{2} \begin{pmatrix} 2 & f_{12} - f_{21} \\ f_{21} - f_{12} & 2 \end{pmatrix}.$$

Then we identify  $\mathbf{T}(\mathbf{F}_t) = \mathbf{R}_t^\dagger$  as the prototype deformation tensor in (11) that gives a (linearized) quadratic energy that measures deviation from rigidity. Triangle-constant local energies  $\|\mathbf{E}_t \mathbf{u}_t - \mathbf{c}_t\|^2$  are then given by

$$\mathbf{E}_t = \frac{1}{2} \begin{pmatrix} 2 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 2 \end{pmatrix} \mathbf{H}_t, \text{ and } \mathbf{c}_t = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 1 \end{pmatrix}.$$

Similar to (13), the global energy matrix  $\mathbf{E}$  is composed block-wise of all  $\mathbf{E}_t$ , and  $\mathbf{c}$  is obtained by stacking all  $\mathbf{c}_t$ . Again, the number of local energy components is  $n = 4$ .

## 5.2 Nonlinear Planar Deformations

The nonlinear as-rigid-as-possible (ARAP) energy is (11) with exact rotations  $\mathbf{T}(\mathbf{F}_t) = \mathbf{R}_t^*$  as deformation gradient prototypes [Igarashi et al. 2005]. The optimization of these energies, which are nonlinear in  $\mathbf{u}$ , is often performed iteratively with alternating linearizations [Sorkine and Alexa 2007; Liu et al. 2008] or by directly using nonlinear solvers [Chao et al. 2010; Lipman 2012]. We focus on the first type of iterative optimization, which is most related to our framework as each step minimizes a quadratic energy. In each iteration  $k$  an intermediate solution  $\mathbf{u}^k$  is improved by computing the closest rotations to the deformation gradients  $\mathbf{F}_t^k$  of  $\mathbf{u}^k$  relative to the initial shape  $\mathbf{x}$ . Here,  $\mathbf{R}_t^*$  of (14) could be used. A numerically more stable variant of this polar decomposition is based on the singular value decomposition (SVD)  $\mathbf{F}_t^k = \mathbf{S} \Sigma \mathbf{T}^T$ , such that  $\mathbf{R}_t^* = \mathbf{S} \mathbf{T}^T$  is the polar decomposition of  $\mathbf{F}_t^k$ . (There are alternative methods to obtain the polar decomposition.) Furthermore, we require the assumption that there is no reflection, i.e., we have  $\det(\mathbf{S} \mathbf{T}^T) = +1$ . The nonlinear computation of optimal rotations is performed locally per triangle and does not involve any regularization. It is the prerequisite for the next step: the definition of a quadratic energy for the  $(k+1)$ -th iteration step. Therefore, the matrices  $\mathbf{R}_t^*$  are used as the target deformation gradients to reconstruct the coordinates of  $\mathbf{u}^{k+1}$  in the least-squares sense. Reconstruction errors are triangle-constant due to the constant gradients of  $\mathbf{u}$ . Using  $\mathbf{F}_t = \mathbf{G}_t \mathbf{U}_t$ , the corresponding integrated piecewise constant reconstruction energy

(11) can be decomposed into component-wise functions  $\mathbf{U}$ :

$$\begin{aligned} E_P(\mathbf{u}^{k+1}) &= \sum_{t \in \mathcal{T}} A_t \|\mathbf{G}_t \mathbf{U}_t^{k+1} - \mathbf{R}_t^*\|_F^2 \\ &= \|\mathbf{G} \mathbf{U}^{k+1} - \mathbf{C}^k\|_{\mathbf{A}_n}^2. \end{aligned} \quad (15)$$

This is a global energy similar to (13) with

$$\mathbf{G} = \left( \sum_{t \in \mathcal{T}} \Lambda_{|\mathcal{T}|, |\mathcal{T}|}^{t,t} \otimes \mathbf{G}_t \right) \mathbf{Q}, \quad \mathbf{C}^k = \sum_{t \in \mathcal{T}} \Lambda_{|\mathcal{T}|, 1}^{t, 1} \otimes \mathbf{R}_t^*, \quad (16)$$

$n = 2$ , and an appropriate replication matrix  $\mathbf{Q}$ . The global gradient operator  $\mathbf{G}$  is sparse with a dimension of  $2|\mathcal{T}| \times |\mathcal{V}|$ , and  $\mathbf{C}$  is a dense  $2|\mathcal{T}| \times 2$  matrix. Again, this energy has the form (8) with  $(\mathbf{E}, \mathbf{C}) = (\mathbf{G}, \mathbf{C}^k)$ , i.e., a matrix norm (7) is used, and regularization (9) is straightforward. Also, note that system factorization only has to be performed once and can be reused in each iteration as both  $\mathbf{G}$  and  $\mathbf{W}_n$  are discretized on the initial mesh  $\mathbf{x}$  and independent of the iteration  $k$ . This means that in every step the solution  $\mathbf{u}^{k+1}$  is obtained from back-substitution.

The particular energy type with  $\mathbf{E} = \mathbf{G}$  representing a discrete gradient operator has a more general interpretation. The discrete energy (15) allows to fit scalar fields  $\mathbf{u}$  on  $\mathcal{M}$  to prescribed gradients  $\mathbf{c}$ . In the ARAP case, these are (component-wise) deformations  $\mathbf{u}$  and prescribed deformation gradients  $\mathbf{R}^*$ . In general, minimizers of this type of energy operator are described by the well-known discrete *Poisson equation* that is here equivalent to (10) for  $\beta = 0$ . Therefore, by using  $\mathbf{W}_n$  instead of  $\mathbf{A}_n$  any Poisson-type energy on meshes (see, e.g., [Pinkall and Polthier 1993; Botsch et al. 2010]) can be regularized using our approach.

## 5.3 Surface Deformations

Not only 2D planar, but also 3D surface-based deformations can be regularized in the same way in our framework. For example, the planar ARAP deformations of the previous section can directly be extended to 3D surface deformations described by  $\mathbf{x}_i, \mathbf{u}_i \in \mathbb{R}^3$  and regularized subsequently. The surface-based ARAP energies can be setup using vertex one-ring transformations as described by Sorkine and Alexa [2007]. We provide an alternative formulation that is more triangle-centric.

For regularization we require  $3 \times 3$  deformation gradients in a common coordinate system. We compute  $\tilde{\mathbf{F}}_t = \mathbf{G}_t \mathbf{U}_t$  using a  $3 \times 3$  extension of the gradient operator (12) for triangles  $t = (i, j, k)$

$$\mathbf{G}_t = \begin{bmatrix} (\mathbf{x}_j - \mathbf{x}_i)^T \\ (\mathbf{x}_k - \mathbf{x}_i)^T \\ \mathbf{n}_t^T \end{bmatrix}^{-1} \begin{pmatrix} -1 & 1 & 0 \\ -1 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix}, \quad (17)$$

where  $\mathbf{n}_t$  is the normalized triangle normal of  $t$  in the original  $\mathbf{x}$  coordinates, such that the gradients of the coordinate functions of  $\mathbf{u}$  lie in the triangle planes of  $\mathbf{x}$  in  $\mathbb{R}^3$ . However, closest rotations to  $\tilde{\mathbf{F}}_t$  are not well-defined yet as the  $\tilde{\mathbf{F}}_t$  are singular in direction  $\mathbf{n}_t$ . One can combine the ARAP local rotation optimization with the handling of this singularity in a single step by using the  $3 \times 3$  SVD  $\tilde{\mathbf{F}}_t = \tilde{\mathbf{S}} \tilde{\Sigma} \tilde{\mathbf{T}}^T$ : the closest 3D rotation to the deformation gradient is  $\mathbf{R}_t^* = \mathbf{S} \mathbf{T}^T$ , where  $\mathbf{S}$  and  $\mathbf{T}$  are derived from  $\tilde{\mathbf{S}}$  and  $\tilde{\mathbf{T}}$  by replacing their column that corresponds to the vanishing singular value in  $\tilde{\Sigma}$  with  $\mathbf{n}_t$  and  $\mathbf{n}_t^k$ , respectively. Here,  $\mathbf{n}_t^k$  is the normalized normal of  $t$  in the deformation  $\mathbf{u}^k$ . Given local rotations optimized this way as target deformation gradients, the global deformation reconstruction in each iteration is analogous to (15) with (16). In fact, the resulting Poisson system is equivalent to (15), except that

now  $n = 3$  and the 3D versions of  $\mathbf{G}_t$  and  $\mathbf{R}_t^*$  are used to setup  $(\mathbf{E}, \mathbf{C}^k)$ . All properties are inherited from the 2D case, and most importantly this way also surface deformations based on 3D ARAP energies can directly be regularized.

#### 5.4 Continuous Deformations

All deformation examples discussed so far are described by a map from the initial coordinates  $\mathbf{x}$  to the coordinates of the deformed solution  $\mathbf{u}$ . This map is obtained directly as the minimizer of an elastic deformation energy. An alternative way to describe planar deformations is a continuous formulations where  $\mathbf{u}_i(s) \in \mathbb{R}^2$  represent time-dependent tangent velocity fields at a specific time  $s$ . Deformations are then obtained by integrating an initial shape  $\mathbf{x}$  along  $\mathbf{u}(s)$  forward in time, giving time-dependent, continuous deformations  $\mathbf{x}(s)$ . This requires evaluation of  $\mathbf{u}(s)$  on different deformed  $\mathbf{x}(s)$ , and vector fields are obtained by optimizing energies in  $\mathbf{u}$ .

Solomon et al. [2011] propose discretized energies that are quadratic in  $\mathbf{u}$  and yield near-isometric planar deformations. These as-Killing-as-possible (AKAP) energies are of the form

$$E_P(\mathbf{u}(s)) = \sum_{t \in \mathcal{T}} A_t \|\mathbf{J}_t + \mathbf{J}_t^T\|_F^2.$$

This energy measures the squared distance of the  $2 \times 2$  vector field Jacobians  $\mathbf{J}_t^T = \mathbf{G}_t \mathbf{U}_t$  from Jacobians of exact isometric vector fields (known as Killing vector fields), which are anti-symmetric. Here, we reuse the gradient operator (12), with the difference that it has to be discretized on the current  $\mathbf{x}(s)$  for each evaluation of the vector field. Vector field Jacobians, which represent a different gradient type, are constant per triangle, and therefore the energy is also local per triangle and fits our setting. The local energies  $\|\mathbf{E}_t \mathbf{u}_t\|^2$  are given by the local  $4 \times 6$  energy operators

$$\mathbf{E}_t = \begin{pmatrix} 2 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 2 \end{pmatrix} \mathbf{H}_t,$$

and the global  $(\mathbf{E}, \mathbf{c})$  are equivalent to (13) using these local operators and  $n = 4$ .

The same formulation holds for the deformation of 3D volumetric shapes. In this case we have Jacobians  $\mathbf{J} \in \mathbb{R}^{3 \times 3}$ , and  $A_t$  is the volume of a tetrahedron.

#### 5.5 Surface Parametrizations

The parametrization of a surface mesh can be considered as the computation of a map from the surface coordinates  $\mathbf{x}_i \in \mathbb{R}^3$  to coordinates  $\mathbf{u}_i = (u_i, v_i)^T \in \mathbb{R}^2$  in the parametrization domain. This map should minimize some type of distortion.

*As-conformal-as-possible.* Discrete near-conformal parametrizations are obtained by satisfying the Cauchy-Riemann conditions  $\nabla u = R^{\pi/2} \nabla v$  in the least-squares sense: this yields the least-squares conformal maps (LSCM) by Levy et al. [2002]. (A different approach that yields identical solutions was proposed simultaneously by Desbrun et al. [2002].) The corresponding discrete and integrated energy can be written as

$$E_P(\mathbf{u}) = \sum_{(i,j,k)=t \in \mathcal{T}} A_t \left\| \mathbf{G}_t (u_i, u_j, u_k)^T - \mathbf{R}_t \mathbf{G}_t (v_i, v_j, v_k)^T \right\|^2$$

(see, e.g., [Botsch et al. 2010]). Here, we use the surface-based gradient operator (17) for gradients in a common 3D coordinate system, and  $\mathbf{R}_t(\cdot) \equiv \mathbf{n}_t \times (\cdot)$  are  $3 \times 3$  rotation matrices performing  $\pi/2$  rotations of vectors in each triangle plane. The corresponding local energies  $\|\mathbf{E}_t \mathbf{u}_t\|^2$  are given by the  $3 \times 6$  operators  $\mathbf{E}_t = [\mathbf{G}_t \quad -\mathbf{R}_t \mathbf{G}_t] \mathbf{P}$

with an appropriate permutation  $\mathbf{P}$  for the coordinate-wise gradient computation. The global energy is described by  $(\mathbf{E}, \mathbf{c} = 0)$ , where the linear operator is setup as in (13) but using  $\mathbf{E}_t$  as described here with  $n = 3$ .

*As-rigid-as-possible.* As shown by Liu et al. [2008], the ARAP energy used in Section 5.2 can also be employed to compute surface parametrizations with low distortions. To perform the local 2D rotation optimization we compute the constant basis functions of a 2D gradient operator  $\mathbf{G}_t$  in the parametrization domain. This can be done by transforming each triangle  $t = (i, j, k)$  separately from 3D to 2D and computing the  $2 \times 3$  operators by

$$\mathbf{G}_t = (\mathbf{S} \mathbf{R}_t [\mathbf{x}_j - \mathbf{x}_i \ \mathbf{x}_k - \mathbf{x}_i])^{-T} \begin{pmatrix} -1 & 1 & 0 \\ -1 & 0 & 1 \end{pmatrix}. \quad (18)$$

Here,  $\mathbf{R}_t$  is the rotation that transforms  $\mathbf{n}_t$  into  $(0, 0, 1)^T$  to align each triangle locally with the  $xy$ -plane, and  $\mathbf{S} = (\begin{smallmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{smallmatrix})$  projects edge vectors into this plane. By transforming each triangle  $t$  by  $\mathbf{S} \mathbf{R}_t$  we essentially obtain *undeformed* 2D reference triangles for the original 3D surface triangles. A parametrization  $\mathbf{u}^k$  at iteration  $k$  then defines  $2 \times 2$  deformation gradients  $\mathbf{F}_t^k = \mathbf{G}_t \mathbf{U}_t^k$ , and closest rotations  $\mathbf{R}_t^*$  can be fitted as before. Then the global Poisson-based reconstruction of  $\mathbf{u}^{k+1}$  is *identical* to (15) and (16), which can be regularized with  $\beta > 0$ . Note that the definition of the global 2D gradient operator  $\mathbf{G}$  (16) by using (18) does not require a continuous reference mesh in the parametrization domain. In fact, the connectivity of the mesh  $\mathcal{M}$  defines continuity of the solution.

## 6. RESULTS

In this section we show experimental results and provide comparisons. If not stated otherwise, boundary constraints are “hard” interpolation constraints, which can be interpreted as pins or handles on vertices. They are rendered as blue spheres. The color-codes visualize the local energies of  $E_P$  per triangle.<sup>3</sup> Note that in general the plain energy cases ( $\beta = 0$ ) correspond to the original results as published, e.g., in [Liu et al. 2008].

Figure 1 shows the effect of our method for different types of 2D deformations for different shapes. We minimize the ASAP, LARAP (Section 5.1), and ARAP (Section 5.2) energies for  $\beta \in \{0, 0.2, 0.4\}$ . The local variation of energy decreases for increasing amount of energy smoothing. The plain energy solutions suffer from high distortion and flipped triangles near the handles for both ARAP energy types. Our regularization corrects these artifacts, and especially the LARAP results show greatly improved quality. Moreover, global self-intersections of plain deformations, which are most notable in the ASAP case, are suppressed in the regularized solutions, although the regularization is not designed to directly prevent this type of artifact.

We show a simple example of a 3D surface deformation minimizing the ARAP energy in Figure 2. The side and the rear (not shown) of the beetle surface were fixed, and a vertex on the engine hood was moved to deform the shape globally. We show results for both “soft” constraints, which only approximate the handle, and interpolating “hard” handle constraints. In both cases the regularized energy yields smooth deformations that respect the particular constraints, whereas the original energy suffers from artifacts that appear near the handle vertex. This experiment backs the claim that our approach also

<sup>3</sup>The additional material contains a video showing further interactive results, a MATLAB reference implementation with a planar deformation demo, as well as all result meshes of the paper with annotated flipped elements.

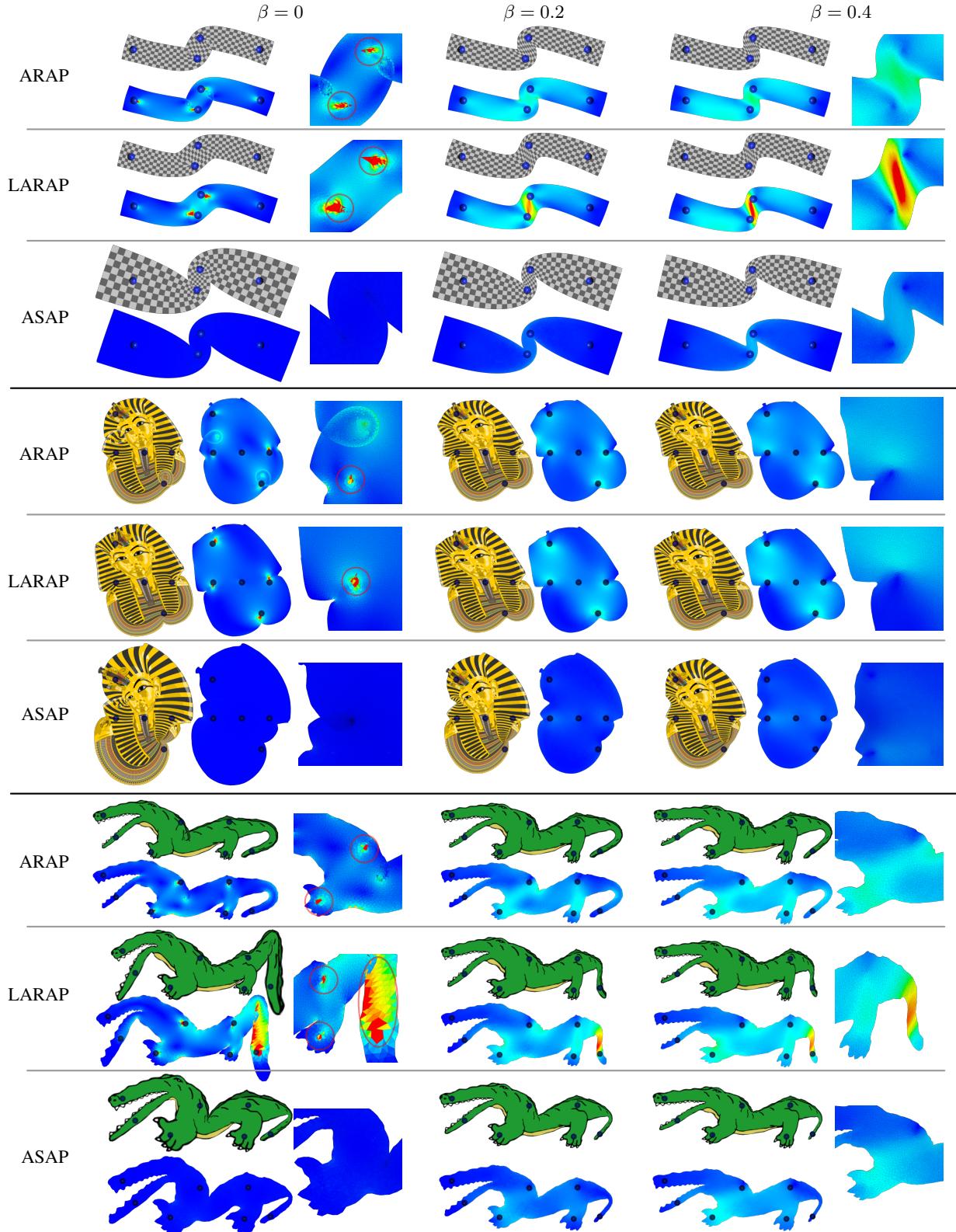


Fig. 1. Regularized 2D deformations. Different 2D shapes are deformed using the different original ( $\beta = 0$ ) and regularized ( $\beta > 0$ ) 2D deformation energies. The color-coded images visualize local energies of  $E_P$  on each triangle (low ●, high ● energy). For  $\beta \in \{0, 0.4\}$  closeups visualize energies at critical regions that contain highlighted high distortions as well as local and global self-intersections.

Pharaoh image courtesy openclipart.org, alligator image from [Jacobson et al. 2011].

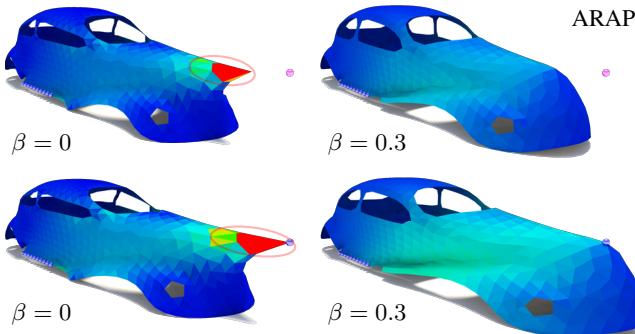


Fig. 2. Regularized ARAP surface deformation. The side and rear (not shown) of the beetle surface are fixed and a vertex on the engine hood is moved to deform the shape such that the ARAP energy is minimized. We show results for both “soft” (•, top) and interpolating “hard” handle constraints (●, bottom). Our regularized solution is artifact-free in both cases.

Beetle mesh courtesy Aim@Shape

works for 3D surface deformations. In particular, ARAP surface deformations show better results.

We apply our regularization to continuous deformations that minimize the AKAP energy in Figure 3. To emphasize total distortions we color-code isometric distortions relative to the original shape. Plain ( $\beta = 0$ ) deformations show notable artifacts as identified by Solomon et al. [2011]. Their filtering method requires the solution of an additional Poisson system that introduces additional distortions and only approximates user constraints. Our method also avoids deformation artifacts, however, and in contrast, we only require a *single* linear system solve and guarantee exact satisfaction of constraints. The tetrahedral eagle model (right) is an example for minimizing smoothed quadratic energies on volumetric domains.

Figure 4 shows examples for parametrizations minimizing LSCM and ARAP energies. We compare plain ( $\beta = 0$ ) and smoothed ( $\beta > 0$ ) energies. The boundary of the beetle parametrization (left) was fixed, and the positions of four *interior* vertices were prescribed. Our regularized LSCM solution has no flipped triangles near the handles. The gargoyle model (right) was cut open and mapped to the plane. (The same cut-open mesh was used in the original work [Liu et al. 2008].) Again, *interior* vertices were fixed. This is an action that is highly relevant in practice for locally “fine-tuning” maps. At the same time this is highly problematic as “spikes” and fold-overs arise quickly, which demands for nonlinear methods. Our ARAP regularization ( $\beta = 0.3$ ) handles this case and provides a valid solution without flipped triangles (see closeups).

We compare our approach to standard Tikhonov regularization in Figure 5. The 2D giraffe shape is deformed, and a parametrization of the 3-balls surface is computed both by minimizing an ARAP energy. Tikhonov regularizers  $E_R^T$  of (5) that penalize first ( $\Gamma_u \equiv \nabla u$ ) and second-order ( $\Gamma_u \equiv \nabla^2 u$ ) variation of the *solution*  $u$  are applied for  $\beta = 0.25$  and  $\beta = 0.5$ . Also, we apply first and second-order Tikhonov regularizers of the displacement field  $d = u - x$  given by the difference of the solution to the initial shape  $x$ . We compare to our regularizer  $E_R$  for  $\beta = 0.25$  (right hand side for each example). First-order Tikhonov regularization of  $u$  yields smoother but undesirably “shrunken” results of high energy. In fact, it even modifies the initial zero-energy solution when handles are not moved at all due to its “problem-independence”. Our proposed method does not show this effect because both, the energy and the problem-specific regularization terms, vanish on the initial pose. Shrinking and modification of the zero-energy solution can be prevented for deformations

by instead regularizing displacement fields  $d$ , but artifacts close to the handles are still not corrected. The results of standard second-order Tikhonov regularization are not very sensitive to the amount of regularization. However, this regularizer is also unable to prevent common artifacts. Not shown are other typical choices of  $\Gamma_u$  like penalizing solution magnitude ( $\Gamma_u \equiv I u$ ): they do not remedy undesired behavior either. This general statement holds similarly for minimizing other energies.

Figure 6 compares our results with the nonlinear bounded distortion (BD-)maps proposed by Lipman [2012]. We consider near-conformal deformations from minimizing BD-LSCM and ASAP energies and near-rigid deformations from minimizing BD-ARAP and ARAP energies. Without regularization ( $\beta = 0$ ) we obtain the undesired artifacts that were pointed out by Lipman (red circles). Adding our regularization  $\beta > 0$  effectively removes artifacts and yields valid shapes without local self-intersections. Interestingly, our solution with the smoother energy distribution for ASAP ( $\beta = 0.25$ ) shows significantly less global deformation compared to BD-LSCM. The BD-ARAP and the regularized ARAP deformation show comparable results: regularized ARAP yields a smoother energy distribution and a thinner neck of the dinosaur. The value  $\beta = 0.085$  is the smallest regularization parameter for which the deformation is artifact-free. Note that there are constraints on *internal* vertices. This leads to a very visible artifact for the unregularized ARAP ( $\beta = 0$ ).

The experiment in Figure 7 shows that the proposed regularization is not only problem-specific but also largely independent of the particular mesh tessellation. We apply an ARAP deformation to three different meshes that all represent the same geometry. Note that this particular shape with thin parts and relatively long boundary is a nontrivial benchmark. The tessellations are challenging as triangle areas vary up to  $1.7 \cdot 10^3\%$ , and the triangle circumcircle to incircle radius ratio is up to 38.8. The results are similar for all tessellations, we show this for two different values of  $\beta$ . This indicates that our discretization in Section 4 is reasonable.

## 7. DISCUSSION

We tested our regularization on a number of typical problems in geometry processing and showed experimental results. Our approach fits problems in this application domain, but it is generally not limited to geometry processing.

*Need for Regularization.* It is noticeable that in particular for these applications, regularization is commonly not considered at all. The reason is probably simple: first, standard Tikhonov regularization is available but it does not fit the problems and yields rather undesired results. Second, most discretized problems are *not ill-posed* in the classical sense, as unique and stable solutions exists, and any regularization might increase the energy  $E_P$  that characterizes the problem. The first argument seems true as indicated by our experiments. This holds also for the very popular Tikhonov-type regularization to prefer “soft” constraints, i.e., a penalty term as regularizer, over “hard” constraints, i.e., constraint interpolation by elimination, in order to suppress or smooth out artifacts near handles. Soft-constraints, however, have their drawbacks: artifacts only appear later but are not effectively suppressed (see Figure 2), and in addition, the solution may “float” in the domain if no point is fixed by a hard-constraint. We refer to, e.g., Solomon et al. [2011], who describe this effect. There are exceptions to the second argument: for instance, minimizing the AKAP energy [Solomon et al. 2011] is an example for a problem that is not ill-posed, though it inherently requires regularization. However, there the standard approach of

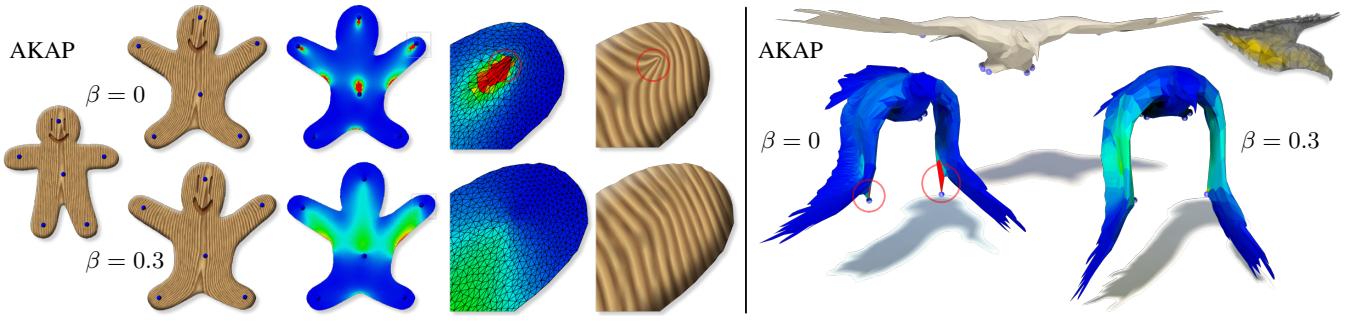


Fig. 3. Regularized continuous deformations. The AKAP energy is minimized and the resulting vector field is integrated for a planar triangle mesh (top) and a volumetric tetrahedral mesh (bottom). The color-coded visualizations show local isometric distortions relative to the rest-pose. Artifacts of the original energies ( $\beta = 0$ ) are highlighted (●).

Woody image from [Jacobson et al. 2011], eagle data from [Chen et al. 2009].

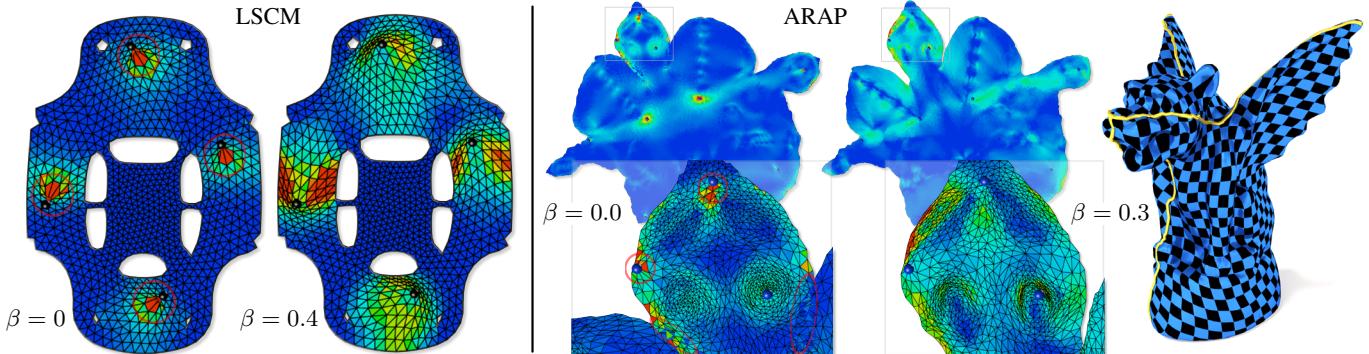


Fig. 4. Regularization of parametrizations. Left: LSCM of the beetle with boundary and four *interior* vertices fixed in the plane. Right: ARAP parametrization of the gargoyle with constraints on *interior* vertices. In contrast to the original solutions our regularized solutions are free of triangle flips (see closeups).

Beetle mesh courtesy Aim@Shape, gargoyle mesh from [Liu et al. 2008].

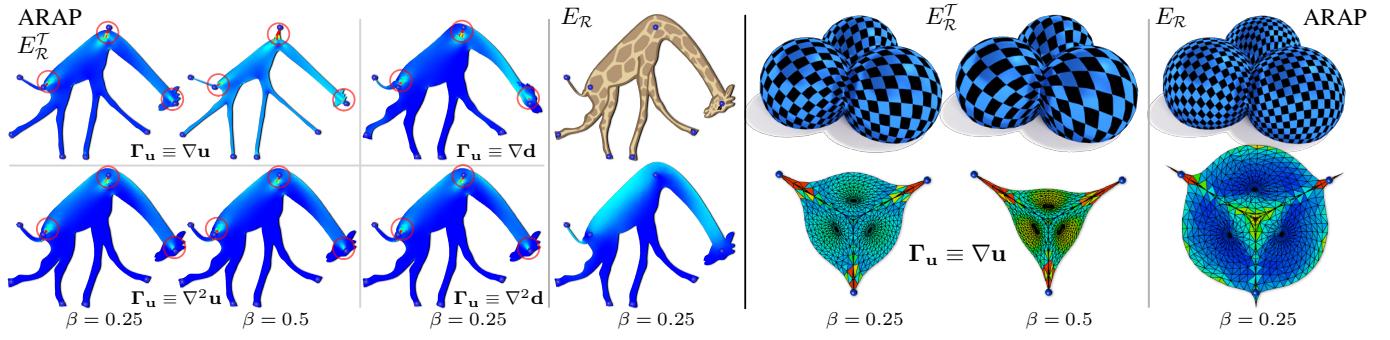


Fig. 5. Comparison to Tikhonov regularization. The ARAP energy is minimized to deform the giraffe (left) and to parametrize the 3-balls surface (right) using different regularizers and  $\beta$  values: standard first ( $\Gamma_u \equiv \nabla \cdot$ ) and second-order ( $\Gamma_u \equiv \nabla^2 \cdot$ ) Tikhonov regularization  $E_R^T$  of the solution  $u$  and displacement field  $d = u - x$  and our regularizer  $E_R$ . Compared to Tikhonov regularization our results show no shrinking behavior and are free of flipped triangles (●).

Giraffe image courtesy openclipart.org, 3-balls mesh from [Liu et al. 2008].

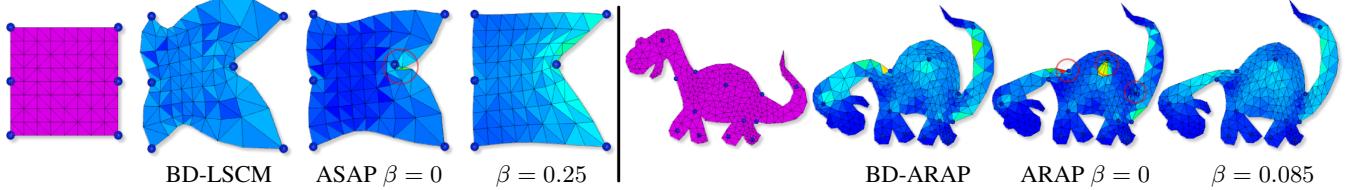


Fig. 6. Comparison with nonlinear BD-maps [Lipman 2012]. Near-conformal (left) and near-rigid (right) deformations of the initial models (●) using bounded distortion mappings and our corresponding regularized energies ( $\beta > 0$ ). Artifacts of the original energies ( $\beta = 0$ ) are highlighted (●).

Square and dino meshes and BD-maps results from [Lipman 2012].

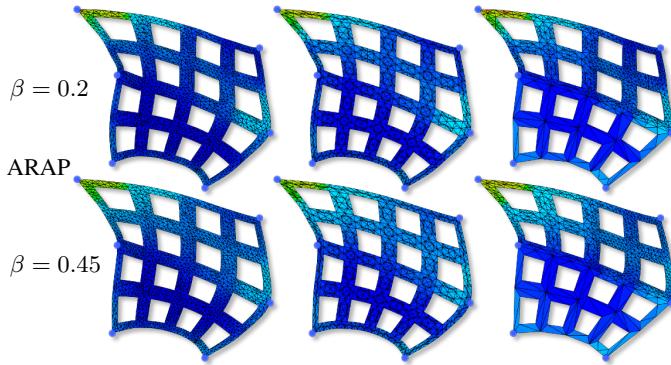


Fig. 7. Mesh dependence. A differently tessellated regular grid is deformed using the same deformation constraints by minimizing the regularized ARAP energy for two  $\beta$  values. Other regularized energies show similar results.

incorporating smoothness of the solution, i.e., standard Tikhonov regularization, would lead to the demonstrated undesirable results. For this reason, Solomon et al. propose an additional Poisson-based smoothing step as a post-process.

In a more general view, the observed artifacts near pointwise constraints are a problem of discretization: in the continuous problem setting, prescribing pointwise constraints in the *interior* of the domain is not a well-posed problem, as solutions of the corresponding first-order variational problems generally ignore these constraints (see, e.g., [Braess 2007]). When discretizing these continuous problems, discrete solutions may exist but can degenerate under different discretizations, leading to noticeable artifacts and the need for regularization.

Hence, we conclude that in comparison to standard problem-independent Tikhonov regularization, our problem-specific approach provides the (more) desirable results: unregularized zero-energy solutions are still zero-energy solutions of the regularized problem. Even more, the regularization enables imposing “hard” interpolation constraints not only on the boundary. Our experiments show that constraints *inside* the domain are possible, at least to some extent. This makes minimizers of regularized quadratic energies competitive with solutions from nonlinear optimization. Of course there is no guarantee that, e.g., triangle flips are avoided.

**Nonlinear Optimization.** Quadratic energies are attractive as models because their minimization is straightforward and numerically efficient. However, the range of problems that can be modeled by quadratic energies is of course limited. In many settings, nonlinear models are required or preferred. A prominent example is imposing general constraints or bounds on variables. This is often done to guarantee properties of the solution, as demonstrated recently by Aigerman and Lipman [2013] and Schüller et al. [2013]. We consider the work by Lipman [2012] as a typical example: a sophisticated nonlinear model guarantees bounded conformal distortion and validity of piecewise linear 2D (deformation) maps. The type of distortion is general, and validity refers to the absence of triangle flips. This is much more than can be expected from a linear setting, i.e., quadratic energy minimization, but it requires the more expensive solution of a nonlinear problem, i.e., one has to invest more time and/or reduce the problem size. Note that in this case the bounds on variables, which restrict the solution space, could be interpreted as a regularization. If a feasible solution exists in this space then it satisfies the guarantees by construction. Having similar guarantees is generally not possible within our setting of

quadratic energy minimization. The comparisons by Lipman reveal these problems, they show up early and frequently for typical deformation tasks. This changes, however, for quadratic energies that are smoothed by our approach. There are still no guarantees, but our experiments show that the regularized solutions get significantly closer to results from nonlinear optimization. And if they break, i.e., they yield an invalid result, this happens relatively late, meaning for extreme deformations imposed by “large” movement of handles.

We conclude that although our regularization cannot provide similar guarantees as nonlinear optimization, minimization in the regularized function space yields competitive results in an efficient way that seem to remain valid for a considerable range of input constraints. This is not possible without or even with standard regularization. In contrast to the mentioned nonlinear methods our regularization is not restricted to deformation-type problems of 2D and 3D shapes, but additionally supports, e.g., surface deformations and vector field-based problems.

**Cost.** The setup of the matrix  $\mathbf{W}_n$  and the regularized normal equations (10) takes more operations than the initial setup (8). More importantly, for sparse operators the new system matrix has more nonzero entries. Our experiments show that there is no significant effect on sparsity of the Cholesky factors  $\mathbf{L}$  and on overall timings compared to having no regularization. Moreover, the overhead is on par with that of standard Tikhonov regularization penalizing second-order variation of the solution.

**Limitations.** Our regularization is restricted to quadratic energies on discrete domains. This is a standard scenario for geometry processing. We are not limited to such scenarios but it may be the case that the penalization of energy variation pays off in particular for this sort of applications. The penalization of variation in the solution may be better suited for other applications. In particular, for general inverse problems a more general regularization based on the (generalized) singular value decomposition with filtering of singular values and the possibility for parameter estimation is generally preferred [Hansen 2010]. However, for geometry processing this is usually less suited as global SVD computations generally turn out to be too expensive.

We show empirically that the minimization of smoothed quadratic energies tends to get close to solutions from nonlinear optimization. There is of course no guarantee for such behavior. In particular, the regularized solutions tend to break later but they will break, i.e., represent an invalid result, at some point for large handle movements. Moreover, there is no a priori optimal choice for  $\beta$ . Although  $\beta$  is not critical for the systems to be solvable, its “optimal” value is certainly problem-dependent, and the applicability of automatic parameter-choice techniques has to be studied. Regularization of quadratic energies is therefore not a substitute for nonlinear models with guarantees, but it may complement these models: the regularized solution can be computed quickly. If it is valid (and satisfies any other criteria) it may be used directly. Otherwise it may serve as an initial guess for a nonlinear solver.

## 8. CONCLUSION

Many problems can be formulated as the minimization of a quadratic energy. We proposed a generic construction of *smoothed* quadratic energies: the input consists of *any* quadratic energy that is integrated in a discrete domain, which is given as a partition of simplices. The output consists of a new energy that incorporates a *problem-specific* regularization. This construction is generic and can be applied in a straightforward way if the problem is given in a standard form (8). Essentially, we can interpret our regularization as a simple change

of the norm that is minimized. The key observation of our approach is that the energy, which characterizes the problem, should *explicitly* contribute to the regularization. This makes our regularization different from standard methods based on Tikhonov regularization: it enforces low energy variation instead of a smooth solution. We applied our regularization to a number of geometry processing applications. Our experiments reveal the advantages of incorporating regularization, and they indicate that our regularizers are appropriate for various energies and fit a range of problems.

## ACKNOWLEDGMENTS

We would like to thank the anonymous reviewers for their thorough and constructive comments and suggestions. We thank Maik Schulze for his help with rendering results and for providing mesh data. The *3-balls* and *gargoyle* meshes are provided by [Liu et al. 2008], the BD-maps results are from [Lipman 2012], the *eagle* data is provided by [Chen et al. 2009], the *beetle* mesh is used courtesy of *Aim@Shape*, and we use image data provided by [Jacobson et al. 2011] and [openclipart.org](http://openclipart.org).

## REFERENCES

- Noam Aigerman and Yaron Lipman. 2013. Injective and Bounded Distortion Mappings in 3D. *ACM Trans. Graph. (Proc. SIGGRAPH)* 32, 4 (2013), 106:1–106:14.
- Marc Alexa and Max Wardetzky. 2011. Discrete Laplacians on general polygonal meshes. *ACM Trans. Graph. (Proc. SIGGRAPH)* 30, 4 (2011), 102:1–102:10.
- P. Alliez, D. Cohen-Steiner, Y. Tong, and M. Desbrun. 2007. Voronoi-based variational reconstruction of unoriented point sets. In *Proc. SGP*. 39–48.
- M. Botsch, L. Kobbelt, M. Pauly, P. Alliez, and B. Levy. 2010. *Polygon Mesh Processing*. AK Peters.
- Mario Botsch and Olga Sorkine. 2008. On Linear Variational Surface Deformation Methods. *IEEE TVCG* 14, 1 (2008), 213–230.
- Dietrich Braess. 2007. *Finite Elements - Theory, Fast Solvers, and Applications in Solid Mechanics* (3rd ed.). Cambridge University Press.
- Isaac Chao, Ulrich Pinkall, Patrick Sanan, and Peter Schröder. 2010. A simple geometric model for elastic deformations. *ACM Trans. Graph. (Proc. SIGGRAPH)* 29, 4 (2010), 38:1–38:6.
- Xiaobai Chen, Aleksey Golovinskiy, and Thomas Funkhouser. 2009. A Benchmark for 3D Mesh Segmentation. *ACM Trans. Graph. (Proc. SIGGRAPH)* 28, 3 (2009), 73:1–73:12.
- Mathieu Desbrun, Mark Meyer, and Pierre Alliez. 2002. Intrinsic Parameterizations of Surface Meshes. *Comput. Graph. Forum (Proc. Eurographics)* 21, 3 (2002), 209–218.
- I. Eckstein, J.-P. Pons, Y. Tong, C.-C. J. Kuo, and M. Desbrun. 2007. Generalized surface flows for mesh processing. In *Proc. SGP*. 183–192.
- Per Christian Hansen. 2010. *Discrete Inverse Problems: Insight and Algorithms*. SIAM.
- J. Hoschek and D. Lasser. 1993. *Fundamentals of computer aided geometric design*. A. K. Peters.
- Qi-Xing Huang, Bart Adams, and Michael Wand. 2007. Bayesian Surface Reconstruction via Iterative Scan Alignment to an Optimized Prototype. In *Proc. SGP*. 213–223.
- Takeo Igarashi, Tomer Moscovich, and John F. Hughes. 2005. As-rigid-as-possible shape manipulation. *ACM Trans. Graph. (Proc. SIGGRAPH)* 24, 3 (2005), 1134–1141.
- Alec Jacobson, Ilya Baran, Jovan Popović, and Olga Sorkine. 2011. Bounded Biharmonic Weights for Real-Time Deformation. *ACM Trans. Graph. (Proc. SIGGRAPH)* 30, 4 (2011), 78:1–78:8.
- Ladislav Kavan, Dan Gerszewski, Adam Bargteil, and Peter-Pike Sloan. 2011. Physics-inspired Upsampling for Cloth Simulation in Games. *ACM Trans. Graph. (Proc. SIGGRAPH)* 30, 4 (2011), 93:1–93:9.
- Bruno Lévy, Sylvain Petitjean, Nicolas Ray, and Jérôme Maillot. 2002. Least squares conformal maps for automatic texture atlas generation. *ACM Trans. Graph. (Proc. SIGGRAPH)* 21, 3 (2002), 362–371.
- Yaron Lipman. 2012. Bounded distortion mapping spaces for triangular meshes. *ACM Trans. Graph.* 31, 4 (2012), 108:1–108:13.
- Ligang Liu, Lei Zhang, Yin Xu, Craig Gotsman, and Steven J. Gortler. 2008. A local/global approach to mesh parameterization. *Comput. Graph. Forum (Proc. SGP)* 27, 5 (2008), 1495–1504.
- Ulrich Pinkall and Konrad Polthier. 1993. Computing Discrete Minimal Surfaces and Their Conjugates. *Experimental Mathematics* 2, 1 (1993), 15–36.
- Christian Schüller, Ladislav Kavan, Daniele Panozzo, and Olga Sorkine-Hornung. 2013. Locally Injective Mappings. *Comput. Graph. Forum (Proc. SGP)* 32, 5 (2013), 125–135.
- Justin Solomon, Mirela Ben-Chen, Adrian Butscher, and Leonidas Guibas. 2011. As-Killing-As-Possible Vector Fields for Planar Deformation. *Comput. Graph. Forum (Proc. SGP)* 30, 5 (2011), 1543–1552.
- Olga Sorkine and Marc Alexa. 2007. As-Rigid-As-Possible Surface Modeling. In *Proc. SGP*. 109–116.
- David M. Strong and Tony F. Chan. 2000. Edge-preserving and Scale-dependent Properties of Total Variation Regularization. In *Inverse Problems*. 165–187.
- Ganesh Sundaramoorthi, Anthony J. Yezzi, and Andrea Mennucci. 2007. Sobolev Active Contours. *Springer IJCV* 73, 3 (2007), 345–366.
- Max Wardetzky, Saurabh Mathur, Felix Kaelberer, and Eitan Grinspun. 2007. Discrete Laplace operators: No free lunch. In *Proc. SGP*. 33–37.

## APPENDIX

We show that the total regularized energy  $E(\mathbf{u})$  in (3) is equivalent to (9), and the induced norm  $\|\cdot\|_{\mathbf{W}_n}$  is well-defined for  $\beta \in [0, 1]$ , i.e.,  $\mathbf{W}_n$  is positive definite. For the residual vector  $\mathbf{r} = \mathbf{E}\mathbf{u} - \mathbf{c}$  we obtain

$$\begin{aligned} E(\mathbf{u}) &= (1 - \beta) \|\mathbf{r}\|_{\mathbf{A}_n}^2 + \beta \|\mathbf{D}_n \mathbf{r}\|_{\mathbf{B}_n}^2 \\ &= (1 - \beta) \mathbf{r}^T \mathbf{A}_n \mathbf{r} + \beta \mathbf{r}^T \mathbf{D}_n^T \mathbf{B}_n \mathbf{D}_n \mathbf{r} \\ &= \mathbf{r}^T ((1 - \beta) \mathbf{A}_n + \beta \mathbf{D}_n^T \mathbf{B}_n \mathbf{D}_n) \mathbf{r} \\ &= \mathbf{r}^T \mathbf{W}_n \mathbf{r} = \|\mathbf{r}\|_{\mathbf{W}_n}^2. \end{aligned}$$

The derivation is similar for matrix residuals  $\mathbf{R} = \mathbf{E}\mathbf{U} - \mathbf{C}$ .

Assuming non-degenerate meshes, the integration matrices  $\mathbf{A}_n$  and  $\mathbf{B}_n$  are diagonal with strictly positive entries, i.e., they are positive definite. The discrete difference operator  $\mathbf{D}_n$  is rank deficient and has a nontrivial kernel spanned by the constant functions. Therefore,  $\mathbf{D}_n^T \mathbf{B}_n \mathbf{D}_n$  is symmetric and positive *semi-definite*, i.e.,  $\mathbf{y}^T \mathbf{D}_n^T \mathbf{B}_n \mathbf{D}_n \mathbf{y} \geq 0$  for  $\mathbf{y} \neq \mathbf{0}$ . Positive definiteness of  $\mathbf{W}_n$  follows from the linearity of sums as for  $\mathbf{y} \neq \mathbf{0}$  and  $\beta \in [0, 1]$  we have  $\mathbf{y}^T \mathbf{A}_n \mathbf{y} > 0$  and

$$\mathbf{y}^T \mathbf{W}_n \mathbf{y} = (1 - \beta) \mathbf{y}^T \mathbf{A}_n \mathbf{y} + \beta \mathbf{y}^T \mathbf{D}_n^T \mathbf{B}_n \mathbf{D}_n \mathbf{y} > 0.$$

Therefore, our norm  $\|\cdot\|_{\mathbf{W}_n}$  is well-defined for  $\beta \in [0, 1]$ .  $\square$

Received July 2013; accepted April 2014