Global parametrization based on Ginzburg-Landau functional

Victor Blanchi, Étienne Corman, Nicolas Ray and Dmitry Sokolov

Abstract Quad meshing is a fundamental preprocessing task for many applications (subdivision surfaces, boundary layer simulation). State-of-the-art quad mesh generators proceed in three steps: first a guiding cross field is computed, then a parametrization representing the quads is generated, and finally a mesh is extracted from the parameterization. In this paper we show that in the case of a periodic global parameterization two first steps answer to the same equation and inherently face the same challenges. This new insight allows us to use recent cross field generation algorithms based on Ginzburg-Landau equations to accurately solve the parametrization step. We provide practical evidence that this formulation enables us to overcome common shortcomings in parametrization computation (inaccuracy away from the boundary, singular dipole placement).

1 Introduction and related work

Meshing is so central in geometric modeling because it provides a way to represent functions on the objects studied (texture coordinates, temperature, pressure, speed, etc.). There are numerous ways to represent functions, but if we suppose that the functions are piecewise smooth, the most versatile way is to discretize the domain of interest. Ways to discretize a domain range from point clouds to block-structured meshes; while the first are really easy to produce, the latter are extremely challenging due to the inherent structure that is very difficult to discover automatically.

In this paper we are interested in the problem of quad mesh generation; although over the past years mesh generation have seen great advances and is now used in

Victor Blanchi
École Normale Supérieure e-mail: `vblanchi@clipper.ens.fr`

Étienne Corman, Nicolas Ray and Dmitry Sokolov
Université de Lorraine, CNRS, Inria, LORIA, 54000 Nancy, France,
e-mail: `name.surname@loria.fr`

production, many challenges remained to be solved for a more practical and efficient use.

Common approaches to quad meshing like paving [3] or Q-morph [9] rely on advancing-front algorithms starting from a constraint, e.g. the boundary, and expanding it to fill the interior. When these advancing fronts meet, they have to be merged; merging two fronts of quads is a challenging task in itself prone to create many local singular vertices. Moreover, due to the propagation process, the mesh topology is fixed near the boundary in the early stage of the algorithm and it is often very difficult to go back on these early choices even if they create terrible quad configuration in the interior of the mesh. All these approaches are based on local decisions that do not consider the global structure of quad/hex meshes, and this is the key reason why quad/hex meshing is so hard. The atomic operation in a triangle/tetrahedral mesh is a simple addition of points; whereas in a quad/hexahedral mesh, it would be the addition quad strips/layers of hexahedra.

While front propagation methods are usually fast and robust (for the 2D case), their output is often not as structured as we would like it to be. Kälberer et al. [5] proposed another type of approach, giving extremely nice looking meshes; unfortunately, it is not guaranteed to be able to mesh all possible domains. The main motivation comes from the observation that good quality quad meshes look like a deformed grid almost everywhere. The idea is to define a deformation of the object such that if the final quad mesh (the result) undergoes this deformation, it matches a unit, axis aligned grid. The direct application of this idea computes this deformation, applies its inverse to the unit grid, and obtains a quad mesh. In practice, it is better to introduce more degrees of freedom by considering a global parameterization instead of a deformation. In this case, parameterizations have some discontinuities that make it possible to represent a much larger family of quad meshes: the deformed grid can be cut and glued to itself in a non-trivial way.

All global parameterization approaches are decomposed into three steps (refer to Fig. 1):

1. *Frame field generation:* this step defines the orientation of the grid at each point of the domain.
2. *Field integration:* this step computes the position and the size of the grid cells aligned with the input orientation field.
3. *Final mesh extraction:* at this step we map the grid onto the original object, thus creating the final quad/hex mesh.

In this paper we focus on the second step, namely the parameterization. Since our approach (periodic global parameterization [11]) is strongly tied to the frame field generation problem, let us review it first.

Frame field generation. A frame is a set of 4 unit vectors $\{z_k\}_{k=0}^3 \subset \mathbb{C}$ which is invariant by a $\pi/2$ rotation. Note that due to its symmetry, a frame can be represented by a single unit vector z such that

$$\{z_k\}_{k=0}^3 = \sqrt[4]{z}\left\{1, e^{i\frac{\pi}{2}}, -1, e^{-i\frac{\pi}{2}}\right\}.$$

Following this definition, we represent frame fields by complex valued fields. We note as $\Re(z)$ and $\Im(z)$ the real and imaginary parts, respectively.

Frame field generation is a simple problem to state: we are looking for the smoothest *unit* field under boundary constraints. Here, smoothest is equivalent to minimizing the Dirichlet energy while the boundary conditions prescribe one of the 4 frame vectors to be aligned with the normal. More formally, we are looking to minimize the following energy:

$$\underset{|z|=1}{\arg\min} \int_\Omega |\nabla z|^2 \quad \text{subject to boundary constraints.} \tag{1}$$

The main challenge is dealing with the unit norm constraint; the very first methods that introduced frame field design for orienting strokes in non-photorealistic rendering [4], represented frame fields by an angle Arg z per vertex, so the unit norm was naturally respected. The optimization was performed by a non-linear solver (BFGS). There is a catch though: Dirichlet energy of a unit frame field is not a reliable measure of quality. While its evaluation on a mesh is finite, in the continuous case the integral diverges, leading to numerically challenging computations. Due to this problem this kind of approaches struggled from bad singularity placement.

Later methods [11, 13, 10] choose to optimize for vector fields directly without the unit norm constraint; the field is normalized in post-processing. This kind of approaches improved greatly the quality of results (and the running times, since only a linear system needs to be solved), but still the geometry of the fields suffered from the unit norm constraint being relaxed. The energy at singular points is well-defined but away from the boundary constraint the frame field is close to zero, making the computation unreliable. Later on, Knöppel et al. [6] proposed to constrain overall norm of the field to be unit (as opposed to the real per-point unit constraint) by solving an eigenvector problem. Fixing the norm of the entire solution instead of the norm of each vector allows to find optimal frame field on closed surfaces, but in presence of boundary constraint the fields are still suboptimal. The latest advances in frame fields [1] proposed to use the Ginzburg-Landau functional to enforce the unity constraint while keeping a well-defined energy. The idea is to redefine the problem (1) as follows:

$$\underset{\text{arbitrary } |z|}{\arg\min} \int_\Omega |\nabla z|^2 + P_\varepsilon(z) \quad \text{subject to boundary constraints.} \tag{2}$$

Here $P(z)$ is a penalty term that enforces the field to have a point-wise unit norm. Beaufort et al. [1] propose to choose $P_\varepsilon(z) = \frac{1}{4\varepsilon^2}(|z|^2 - 1)^2$; this problem is well-defined, and can be solved by Newton iterations. As ε tends to zero, the field converges to a unit-norm field, while minimizing the field variation. Viertel et al. [17] suggested another optimization scheme for Eq. (2) based on heat diffusion. Both methods yield similar results in term frame field quality. It is to be noted that, while these approaches improve the geometry of frame fields in some cases, the problem is still non convex, and does not provide guarantees of optimality.

Parametrization. The parameterization step consists in integrating the frame field, while imposing integer constraints at the boundary and singular points. More precisely, one needs to find two scalar functions [5] whose gradients are as close as possible to the input frame field under the integer constraints. When succeeding, these approaches provide impressive results. There are, however, many limitations. The frame field may not be integrable in the sense that is does not locally correspond to the gradient of a scalar function. This problem can be mitigated by asking the frame field to be curl-free [11, 16, 15]. However this does not prevent a major problem: some frame fields are not consistent with any quadrangulation. Top left image of Fig. 1 provides an example: the radial frame field (without any singularities!) cannot be integrated directly; the only way to compute a parameterization is to insert a dipole (index 1/4 and -1/4) in the scalar fields.

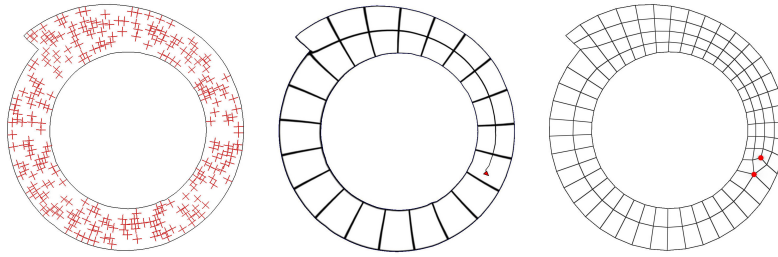**Fig. 1** Quad meshing *via* global parameterization. **Left:** a cross field prescribing the orientation of the elements. **Middle:** a parameterization prescribing the size and the position of the elements to place. Note that it is *not* a quad mesh, it is a unit grid texture image mapped to the triangle mesh. In this case it is a periodic global parameterization, the red triangle shows a singularity of the parameterization. **Right:** a quad mesh extracted from the parameterization. The singularity of the parameterization corresponds to a dipole (a pair of vertices of valence 3 and 5, shown in red).

Getting a *bijective* parametrization for quad-meshing is an open problem. The most advanced methods follow the path of [8] by computing a *motorcycle graph* which is a partition of the surface into quads with possible T-junctions. This intermediate state, in between the full quad mesh and the triangle mesh, allows the authors to formulate the targeted grid aligned parameterization into a mixed-integer problem whose variables are subdivisions or collapses of the T-mesh. Even though this is currently the most robust method available, each step has its practical shortcoming. For instance, extracting motorcycle graphs from a frame field is a notoriously challenging task as it requires to trace non-intersecting streamlines [8, 12]. The intermediate T-mesh has many T-junctions which must be resolved either by collapsing the edge into a quad or by placing a dipole of singularities. In either case it requires to robustly solve a mixed-integer problem which is often NP-complete.

In this landscape algorithms in the wake of Periodic Global Parametrization (PGP) [11, 16] stand aside. The main idea is to consider quad mesh edges as zeros of two oscillators integrating orthogonal branches of the frame. Thus, all integer

constraints are built in the algorithm by design and extracting a mesh does not require to compute streamlines.

The optimization problem involved in PGP is nearly identical to the one for computing frame field: a modified Dirichlet energy with Dirichlet boundary conditions. Thus, similar challenges regarding unit norm constraints and singularity placement arise. The original paper [11] solves the optimization problem by relaxing the unit norm constraint, leading to a nearly zero solution away from the boundary.

Based on the frame field generation experience, we propose an optimization strategy based on the Ginzburg-Landau functional [2]. It correctly specifies the oscillators away from constraints, creates more regular quads and improves the quality of the mesh near singularities.

N.B. For the sake of clarity, we focus on integrating *singularity-free* frame fields. PGP can deal with singular fields through a quad-covering of the surface [11], but we found it unnecessary to show the potential of the Ginzburg-Landau functional in the context of field integration.

2 Periodic global parametrization

In this paper we mesh a planar domain $\Omega \subset \mathbb{R}^2$. In order to simplify the presentation, our input frame fields are without singularities: it allows us to extract two continuous orthogonal vector fields from the input frame field; we integrate these vector fields independently one from another. In this section the vector field to be integrated is denoted as V and $\partial\Omega^v$ denotes the subset where V is orthogonal to the boundary.

2.1 PGP basics

In the standard quad meshing pipeline the edges of the resulting mesh are orthogonal to one of the frame field directions. The goal of the integration step is to compute a global parametrization whose integer level sets respect this constraint. The most basic scheme [5], which laid ground for many quad meshing methods, is to optimize for parameters (u, v) such that their gradients are equal to one of the directions of the frame. At the same time integer constraints are imposed in order to conform the boundary of the domain with the boundary of the quad mesh.

Another possibility is to look for a periodic function. Indeed, the periodic function $z(p) = e^{2\mathrm{i}\pi\theta(p)}$ has clearly identified integer level sets which can be easily aligned with the boundary. A level set of z is orthogonal to the vector field V whenever V is equal to the direction of oscillation $\nabla\theta$. Thus taking the gradient of z, the periodic function must satisfy [11, 7]:

$$\nabla z = 2\mathrm{i}\pi V z.$$

By specifying the norm of V, we determine the speed of oscillation and thus the size of the quads.

As the mesh must conform to the boundary of the domain, we have to enforce the Dirichlet boundary constraint $z = 1$. Moreover z has pointwise unit norm over the domain. So integrating the vector field V with a global periodic function amounts for solving the following optimization problem:

$$\begin{aligned} \min_{z \in \mathbb{C}} \quad & \tfrac{1}{2} \int_\Omega |\nabla z - 2\mathrm{i}\pi V z|^2 \\ \text{s.t.:} \quad & z = 1 \text{ on } \partial\Omega^v, \\ & |z| = 1 \text{ on } \partial\Omega. \end{aligned} \tag{3}$$

The optimization problem in Eq. (3) is non-convex due to the unit norm constraint. The authors of original paper [11] simply removed this constraint altogether to recover a quadratic optimization similar to a Laplacian smoothing operation. Note that if not for the boundary constraints, the solution of Eq. (3) would be equal to zero everywhere. So a common issue with this approach is that far away from the constraint the norm of z is numerically close to zero making the integration unreliable (see Fig. 2 top row). The Ginzburg-Landau functional is a sound way of accurately solving Eq.(3).
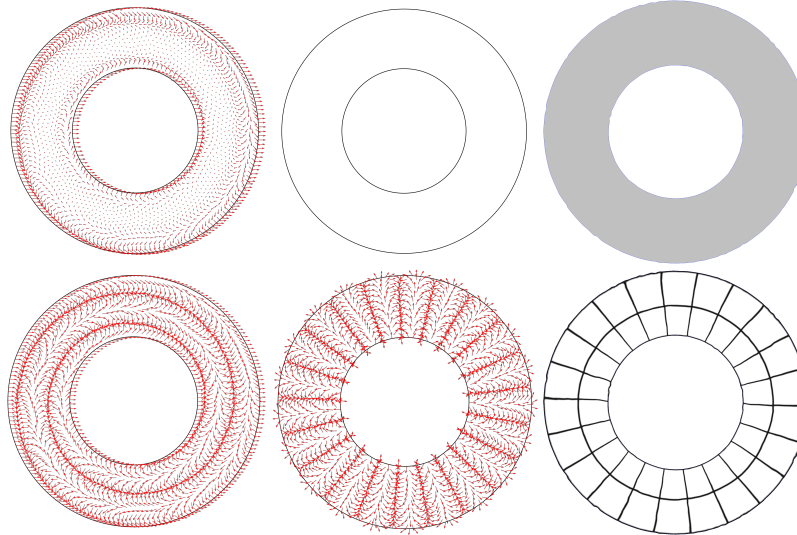

Fig. 2 Quad meshing *via* Periodic Global Parametrization. **Left:** radial field integration. **Middle:** tangent field integration. **Right:** quad mesh extracted from the parameterization. On the **top row** the integration with PGP causes a drastic norm reduction in the integration of the radial field and the tangent field integration simply output a everywhere vanishing field. In comparison, our Ginzburg-Landau regularization on the **bottom row** yield a unit norm vector field on both direction and a valid quad mesh can be extracted.

2.2 Ginzburg-Landau to the rescue

As mentioned earlier, Eq. (3) is a non-convex optimization problem because of the unit norm constraint. Recent work on frame field generation used the Ginzburg-Landau functional to deal with such a constraint. In practice this strategy allows for a better placement of singularities [1]. Note that our problem (Eq. (3)) is very similar to the frame field generation problem (Eq. (2)); following this idea we propose to use a Ginzburg-Landau functional for periodic global parametrization:

$$z_\varepsilon = \arg\min_{z\in\mathbb{C}} \; E(z) + P_\varepsilon(z) = \tfrac{1}{2}\int_\Omega |\nabla z - 2i\pi V z|^2 + \tfrac{1}{4\varepsilon^2}\int_\Omega \left(|z|^2 - 1\right)^2$$
$$\text{s.t.: } z = 1 \text{ on } \partial\Omega^v. \tag{4}$$

Theoretical results [14] demonstrated that as ε goes to zero, the complex function z tends to be a unit complex number everywhere except at isolated singular points. Asymptotically the singular points have integer indexes and are placed in a way that minimizes the overall field curvature.

In the rest of the paper we show how to discretize our problem with standard finite elements and solve the optimization problem with a Newton method for decreasing value of ε.

2.3 Newton method

In order to solve the optimization problem of Equation (4), we use a modified Newton method. More precisely, a local extremum of the energy is reached whenever z_ε is a stationary point of the energy i.e. $\nabla E(z_\varepsilon) + \nabla P_\varepsilon(z_\varepsilon) = 0$. Thus, after explicitly differentiating E and P_ε, the oscillator z_ε must satisfy the non-linear PDE:

$$\begin{cases} \nabla E(z_\varepsilon) + \nabla P_\varepsilon(z_\varepsilon) = 0 & \text{on } \Omega, \\ z_\varepsilon = 1 & \text{on } \partial\Omega^v, \end{cases} \tag{5}$$

where the gradients are explicitly given by:

$$\begin{cases} \nabla E(z_\varepsilon) = -\Delta z + 4\pi^2 |V|^2 z + i\pi \left(\langle \nabla z, V\rangle + \operatorname{div}(Vz)\right) \\ \nabla P_\varepsilon(z) = \tfrac{1}{\varepsilon^2} z \left(|z|^2 - 1\right) \end{cases} \tag{6}$$

For each value of ε, we approximate the solution of (5) with Newton iterations. To realize this scheme, we need an expression of the Hessian matrix as a function of point z. The closed-form expression of the Hessian is easier to read as a matrix applied to the vector $\left(\mathfrak{R}(z)\; \mathfrak{I}(z)\right)^\top$ containing the real and imaginary part of the complex:

$$\begin{cases} H_E(z)h = -\Delta h + 4\pi^2 |V|^2 h + i\pi \left(\langle \nabla h, V \rangle + \text{div} \, (Vh) \right) \\ H_{P_\varepsilon}(z)h = \frac{1}{\varepsilon^2} \begin{pmatrix} 3\Re(z)^2 + \Im(z)^2 - 1 & 2\Im(z)\Re(z) \\ 2\Im(z)\Re(z) & 3\Im(z)^2 + \Re(z)^2 - 1 \end{pmatrix} \begin{pmatrix} \Re(h) \\ \Im(h) \end{pmatrix} \end{cases} \tag{7}$$

Newton iterations are well-defined only when the Hessian is positive definite. The first term $E(z)$ of the energy in Equation (4) is convex thus its Hessian is positive. The second term P_ε constraining the unit norm is non-convex and its Hessian is negative near 0. In this case it is a common practice to approximate the Hessian with a positive definite matrix by removing all negative terms from the Hessian (7):

$$\tilde{H}_{P_\varepsilon}(z)h = \frac{1}{\varepsilon^2} \begin{pmatrix} 3\Re(z)^2 & 2\Im(z)\Re(z) \\ 2\Im(z)\Re(z) & 3\Im(z)^2 \end{pmatrix} \begin{pmatrix} \Re(h) \\ \Im(h) \end{pmatrix} \tag{8}$$

Therefore, for a fixed ε we compute the sequence of oscillator z_ε^n satisfying:

$$\begin{cases} \left(H_E(z_\varepsilon^n) + \tilde{H}_{P_\varepsilon}(z_\varepsilon^n) \right) h_\varepsilon^{n+1} = -\nabla E(z_\varepsilon^n) - \nabla P_\varepsilon(z_\varepsilon^n) \\ z_\varepsilon^{n+1} = z_\varepsilon^n + h_\varepsilon^{n+1} \end{cases}, \tag{9}$$

until convergence to a stationary point.

2.4 Discretization

We discretize Eq. (9) with standard first order finite elements. The oscillator z is a complex per vertex and the vector field V is encoded as a two dimensional vector per vertex. Both are linearly interpolated on each triangle with "hat" functions φ (Fig. 3). On the triangle (ijk) the interpolation, denoted with superscript ^{ijk}, reads: $z^{ijk} = z_i\varphi_i + z_j\varphi_j + z_k\varphi_k$.
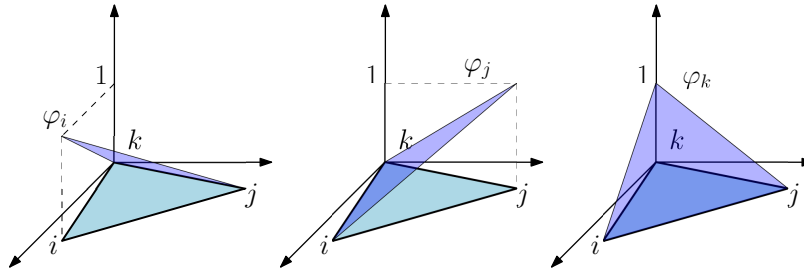

Fig. 3 The three local FEM basis functions on the element ijk.

The Hessian (8) are square complex matrices of size the number of vertices. They are assembled by accumulated the 6×6 elementary matrices computed on an element Ω_{ijk}, given by:

$$
\begin{cases}
H_E^{ijk} = \int_{\Omega_{ijk}} -\langle \nabla\varphi_n, \nabla\varphi_m \rangle + 4\pi^2 |V^{ijk}|^2 \varphi_n\varphi_m + \\
\qquad\qquad + i\pi \left(\langle \nabla\varphi_n, V^{ijk} \rangle + \operatorname{div} \left(V^{ijk}\varphi_n \right) \right) \varphi_m \\
H_{P_\varepsilon}^{ijk} = \frac{1}{\varepsilon^2} \begin{pmatrix} 3 \int_{\Omega_{ijk}} \Re(z^{ijk})^2 \varphi_m\varphi_n & 2 \int_{\Omega_{ijk}} \Im(z^{ijk})\Re(z^{ijk})\varphi_m\varphi_n \\ 2 \int_{\Omega_{ijk}} \Im(z^{ijk})\Re(z^{ijk})\varphi_m\varphi_n & 3 \int_{\Omega_{ijk}} \Im(z^{ijk})^2 \varphi_m\varphi_n \end{pmatrix}
\end{cases}
$$

for $m, n = 1, 2, 3$. Similarly, Equation (5) is turned into a vector of complex of the size number of vertex. The elementary vector of size 3 on an element Ω_{ijk} are given by:

$$
\begin{cases}
\nabla E^{ijk} = \int_{\Omega_{ijk}} -\langle \nabla z^{ijk}, \nabla\varphi_m \rangle + 4\pi^2 |V^{ijk}|^2 z^{ijk}\varphi_m + \\
\qquad\qquad + i\pi \left(\langle \nabla z^{ijk}, V^{ijk} \rangle + \operatorname{div} \left(V^{ijk} z^{ijk} \right) \right) \varphi_m \\
\nabla P_\varepsilon^{ijk} = \frac{1}{\varepsilon^2} \int_{\Omega_{ijk}} z^{ijk} \left(|z^{ijk}|^2 - 1 \right) \varphi_m
\end{cases}
$$

with $m = 1, 2, 3$.

3 Results

This article focuses on the parametrization step of the quad meshing pipeline, thus the results shown in Fig. 4 show a unit grid texture image applied to the domain, we do not extract actual quad meshes.

For the sake of clarity, we presented an algorithm for integrating *non-singular* frame fields over a *flat* 2D domain. While it is perfectly possible to compute a periodic global parameterization of a 3D triangulated surface with Ginzburg-Landau functional, as well as to incorporate the frame field singularities in the computation, the notations become cumbersome and we prefer to avoid it.

For Figure 1, the vector fields are obtained by solving Eq. (2) as specified in [1]. For the other examples (Fig. 4 and Fig. 5), the vector fields are computed as two independent orthogonal unit vector fields U and V. First, we partition the boundary in two complementary sets based on the normal orientation. If the normal is close to the x-axis, the point belongs to $\partial\Omega^u$, but if it is closer to the y-axis, it belongs the set $\partial\Omega^v$. Second, we find the smoothest non-singular unit vector field U normal to the boundary on $\partial\Omega^u$ and tangent to the boundary on $\partial\Omega^v$ by smoothly interpolating the angle of the vector. The vector field V is obtained by $90°$ rotation of U.

The vector fields U, V are then integrated separately by doing the Newton iterations of Eq. (9) for decreasing values of ε. The boundary conditions naturally arise from the construction of the vector fields i.e. $U = 1$ on $\partial\Omega^u$ and $V = 1$ on $\partial\Omega^v$.

The frame field on Fig. 1 exhibits a limit cycle making it non-integrable. Thus standard integration algorithms often fail to output a valid quad mesh. Our method is able to correctly place a singularity (middle image) which can be turn it a quad mesh (right image). Frame fields used for our results are singularity-free, yet our algorithm is able compute smooth parametrizations with a minimal amount of parameterization singularities as shown in Fig. 4. In Fig. 5, we compare our results with a naive implementation of PGP, simply solving Eq. (3) as a quadratic problem by removing
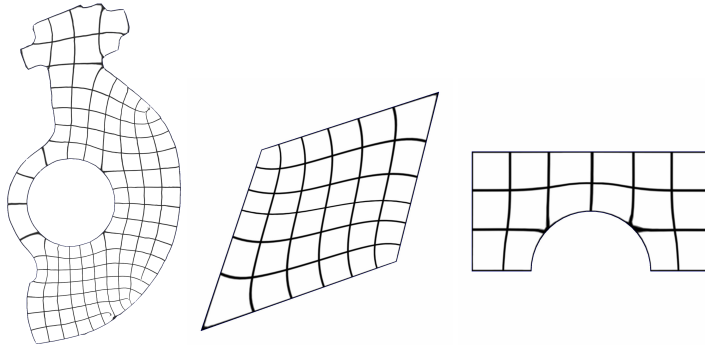
Fig. 4 Examples of parametrization obtained with our resolution of periodic global parametrization with Ginzburg-Landau functional. Note that the algorithm tries to limit the number of singularities and optimally places unavoidable dipoles.

the unit norm constraint. It is easy to see that the parameterization obtained with the Ginzburg-Landau functional is more regular and presents less singularities. Moreover, in the some drmatic cases, omitting the unit norm constraint makes PGP unable to output valid parametrization (see Fig. 2)

4 Conclusion

The main goal of this article is to underline the similar nature between frame field generation and periodic global parameterizations. Both problems can be solved by (almost) the same set of equations using the Ginzburg-Landau functional. This article brings a solid theoretical basis that allows to solve accurately the periodic global parameterization problem, making the problem well-posed and leading to an automatic and optimal singularity placement in the parametrization.

References

1. Beaufort, P.A., Lambrechts, J., Henrotte, F., Geuzaine, C., Remacle, J.F.: Computing cross fields a pde approach based on the ginzburg-landau theory. Procedia engineering **203**, 219–231 (2017)
2. Bethuel, F., Brezis, H., Hélein, F., et al.: Ginzburg-landau vortices, vol. 13. Springer (1994)
3. Blacker, T.D., Stephenson, M.B.: Paving: A new approach to automated quadrilateral mesh generation. International Journal for Numerical Methods in Engineering **32**(4), 811–847 (1991). DOI 10.1002/nme.1620320410. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/nme.1620320410
4. Hertzmann, A., Zorin, D.: Illustrating smooth surfaces. In: PROCEEDINGS OF SIGGRAPH 2000, pp. 517–526 (2000)
5. Kälberer, F., Nieser, M., Polthier, K.: Quadcover-surface parameterization using branched coverings. In: Computer graphics forum, vol. 26, pp. 375–384. Wiley Online Library (2007)
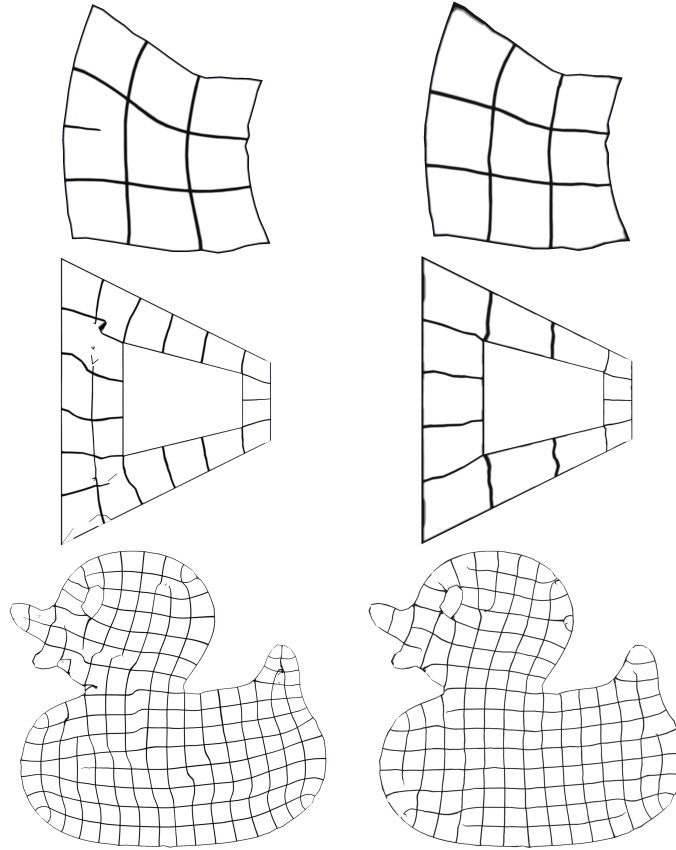
Fig. 5 Comparison between one step of periodic global parametrization (left) and our periodic global parametrization with Ginzburg-Landau functional (right). Note that our algorithm removes unnecessary singularities and produces more regular quads.

6. Knöppel, F., Crane, K., Pinkall, U., Schröder, P.: Globally optimal direction fields. ACM Trans. Graph. **32**(4) (2013)
7. Knöppel, F., Crane, K., Pinkall, U., Schröder, P.: Stripe patterns on surfaces. ACM Transactions on Graphics (TOG) **34**(4), 1–11 (2015)
8. Myles, A., Pietroni, N., Zorin, D.: Robust field-aligned global parametrization. ACM Transactions on Graphics (TOG) **33**(4), 1–14 (2014)
9. Owen, S.J., Staten, M.L., Canann, S.A., Saigal, S.: Q-morph: an indirect approach to advancing front quad meshing. International Journal for Numerical Methods in Engineering **44**(9), 1317–1340 (1999). DOI 10.1002/(SICI)1097-0207(19990330)44:9<1317::AID-NME532> 3.0.CO;2-N. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/%28SICI% 291097-0207%2819990330%2944%3A9%3C1317%3A%3AAID-NME532%3E3.0.CO%3B2-N
10. Palacios, J., Zhang, E.: Rotational symmetry field design on surfaces. ACM Trans. Graph. **26**(3) (2007). DOI 10.1145/1276377.1276446. URL http://doi.acm.org/10.1145/1276377. 1276446
11. Ray, N., Li, W.C., Lévy, B., Sheffer, A., Alliez, P.: Periodic global parameterization. ACM Transactions on Graphics (TOG) **25**(4), 1460–1485 (2006)

12. Ray, N., Sokolov, D.: Robust polylines tracing for n-symmetry direction field on triangulated surfaces. ACM Transactions on Graphics (TOG) **33**(3), 1–11 (2014)
13. Ray, N., Vallet, B., Alonso, L., Levy, B.: Geometry-aware direction field processing. ACM Trans. Graph. **29**(1), 1:1–1:11 (2009). DOI 10.1145/1640443.1640444. URL `http://doi.acm.org/10.1145/1640443.1640444`
14. Rivière, T., Given, M.c., Harpes, P.: Asymptotic analysis for the ginzburg-landau equations (1997)
15. Sageman-Furnas, A.O., Chern, A., Ben-Chen, M., Vaxman, A.: Chebyshev nets from commuting polyvector fields. ACM Transactions on Graphics (TOG) **38**(6), 1–16 (2019)
16. Sokolov, D., Ray, N., Untereiner, L., Lévy, B.: Hexahedral-dominant meshing. ACM Trans. Graph. **35**(5) (2016). DOI 10.1145/2930662. URL `https://doi.org/10.1145/2930662`
17. Viertel, R., Osting, B.: An approach to quad meshing based on harmonic cross-valued maps and the ginzburg–landau theory. SIAM Journal on Scientific Computing **41**(1), A452–A479 (2019)