*Proceedings of the IASS 2024 Symposium*
*Redefining the Art of Structural Design*
August 26-30, 2024, Zurich, Switzerland
Philippe Block, Giulia Boller, Catherine DeWolf,
Jacqueline Pauli, Walter Kaufmann (eds.)

# Solving bilinear equations to align conjugate curvatures and stress directions in the NURBS-based form finding of shells

Masaaki MIKI[*] and Toby MITCHELL[a]

[*] The University of Tokyo
Komaba 3-8-1, Meguro-ku, Tokyo, Japan
masaakim@g.ecc.u-tokyo.ac.jp
[a] Thornton Tomasetti, Chicago, United States of America

## Abstract

Metal-glass grid shells are increasingly being constructed worldwide. However, the efficiency of their construction can be improved from both structural and assembly viewpoints. Many studies have noted that aligning the principal stress and principal curvature nets enables the design of grid shell structures that are both structurally efficient and easy to construct. This is because principal stress trajectories give a bending-free grid, and a principal curvature net gives a flat panelization. We have previously presented a novel form-finding method for shell structures that employs NURBS to represent the geometry of a smooth shell[1]; however, its applications are limited to reinforced concrete shells. The present study considers how an alignment condition can be incorporated into this previous method because, when two nets are aligned on a form-finding result represented by smooth surfaces, the smooth surfaces can be further interpreted into a grid shell. As noted in [2], it is also possible to select conjugate stress and curvature nets. Therefore, we discuss the alignment of conjugate nets as well. Our contribution lies in representing the alignment condition through a bilinear symmetric partial differential equation, which has the same form as the equilibrium equation of general membrane shells (Pucher's equation). The newly introduced alignment condition and the equilibrium equation can be simultaneously solved using a combination of NURBS surfaces and the variable projection method used to solve the equilibrium equation in [1]. In addition to the introduction of the alignment condition, we also experimentally introduce a boundary-shape optimization strategy, which is anticipated to 'correct' the impossible boundary shapes and guarantees the existence of an Airy stress function.

**Keywords**: Airy's stress function, form finding, shell structure, bilinear partial differential equation, variable projection method

## 1. Previous Works

In architectural design, *form-finding* is a process of finding structurally efficient shapes. Often, this entails finding special curves or surfaces that can withstand gravity with no bending action but with only axial or in-plane stresses. The form-finding of *shells* is basically analogous to that of a catenary arch: hanging a gypsum-soaked fabric, drying it until it solidifies, and flipping it gives a reinforced concrete shell shape that can withstand its self-weight with pure-compression stresses (e.g., Isler's shell). While this inverted hanging experiment can be easily simulated digitally (e.g., [3, 4, 5, 6, 7]), the form-finding of mixed tension-compression membrane shells is known to be a very difficult problem to solve. This is because a mixed tension-compression stress tensor makes the equilibrium problem hyperbolic, and the boundary problems of hyperbolic problems are known to be very difficult to solve. Specifically, boundary information propagates through characteristic lines, which are the asymptotic lines of a stress function in the context of form-finding of shells; thus, compatible boundary values that do not contradict each other between two ends of characteristic lines must be found first. Structural engineers are usually not trained to solve hyperbolic boundary value problems; the majority of the problems they solve in their daily work are elliptic, in which any collection of boundary values determines a unique solution. Thus, very few studies have investigated mixed tension-compression form-finding (e.g., [8, 9, 10]).

In 2022, we developed a stable and robust computational method for solving the mixed tension-compression form-finding problem [1]. We noted that Chiang et al. [11] were the first to develop a similar method; however, their method did not converge easily or had a poor convergence rate. We concluded that the variable projection (VarPro) method [12, 13] proposed more than 40 years ago specifically for bilinear problems works very well because the equilibrium problem in this type of form-finding problem is a boundary value problem of a bilinear second-order partial differential equation (PDE).

While our previous method was successful, its extensibility remained questionable because VarPro is designed to solve only bilinear problems, and few bilinear problems are encountered in practice. Moreover, while this method can solve a continuum shell form-finding problem correctly, it is limited to continuum equilibrium, and thus, its application is limited to reinforced concrete shells.

Nowadays, metal-glass grid shells are increasingly being constructed worldwide. To use the form-finding result as the basis for the grid shell geometry, ease of construction must be considered; otherwise, too many bent glass panels and complex detailing of joints may be required. The current study is a continuation of our 2022 study and develops a new bilinear second-order PDE called an alignment condition. It allows the design of grid shells by ensuring that the conjugate stress and conjugate curvature nets are aligned on a continuum shell form-finding solution. Thus, a continuum shell can be interpreted as a grid shell that can simultaneously be bending free and covered by planar quadrilateral glass panels. Thus, the continuum shell form-finding solution can be used to design a metal-glass grid shell that is structurally efficient and easy to construct.

The new condition can be solved using VarPro in combination with the original bilinear equilibrium equation. Thus, our extension simultaneously ensures that grid shells are covered by flat quadrilateral glass panels and are bending-free. This is a preliminary study that continues our paper recently accepted by Siggraph 2024, which is scheduled to be published in *Transactions on Graphics* in July [14]. We have further developed and tested the method through many example problems not considered herein. Interested readers can refer to [14]. In addition to introducing the alignment condition, as bonus content, we also experimentally introduce a boundary-shape optimization strategy, which is anticipated to "correct" impossible boundary shapes and guarantee the existence of an Airy stress function.

## 2. Introduction

Previous studies have already extensively discussed the alignment of principal stress and curvature directions in shells [15, 16]. Our method differs from previous ones mainly in that it solves two continuum bilinear PDEs on NURBS surfaces and never explicitly computes two nets. The nets are computed in a post-process after a form-finding solution is obtained. This is significant because other methods directly optimize quad meshes, although the eventual meshes are not known in advance. The two bilinear PDEs constitute an equilibrium equation, *Pucher's equation*, and our new PDE or alignment condition. We have already developed the VarPro computational method for solving the equilibrium equation [1], which is a bilinear second-order PDE. Because the alignment condition has the same bilinear form as the equilibrium equation, it can easily be solved within the existing computational framework. Moreover, our alignment condition can cover conjugate directions alignment as well.

Below, we briefly summarize our proposed alignment condition and present a few variations that provide different alignments.

## 3. Alignment conditions

### 3.1. Alignment of two real symmetric 2 x 2 matrices

The heart of the idea of the alignment condition is that the condition in which the eigenvectors of two real symmetric $2 \times 2$ matrices point to the same direction is equivalent to the condition in which the product of the two matrices is symmetric. This can be generalized to cases with two real symmetric $2 \times 2$ matrices $A$ and $B$, and one real symmetric positive (semi)definite matrix $E$. When $AEB$ is symmetric, one can finds a pair of vectors simultaneously conjugate with $A$ and $B$. This pair is also conjugate with $e = R_{90}ER_{90}^T$.

### 3.2. Various alignment conditions

*3.2.1. Lines-of-curvature alignment*

Using the aforementioned finding, we observe that the following condition can align principal stress (principal stress trajectories) and principal curvature nets (lines of curvature):

$$h\{g^{jk}\}H = \mathrm{sym}, \tag{1}$$

where $g^{ij}$ is the inverse of the metric tensor (the first fundamental form), $h$ is the Hessian of an Airy stress function (i.e., 90-degrees rotated stress tensor), and $H$ is the Hessian of the height function of the shell (i.e., curvature). Imposing this condition results in the strict alignment of the lines of curvature and principal stress trajectories. Notably, the obtained grid is orthogonal and we have no control over the actual directions of the principal directions.

As we are computing everything using NURBS surfaces, those matrices are components of tensors on curvilinear coordinate parameters. in the following, we denote them as $(\theta^1, \theta^2)$. To compute the Hessians, we must use the covariant derivatives instead of partial derivatives. The covariant derivatives can be computed as

$$\nabla_{ij} f = \partial_i \partial_j f - \Gamma_{ij}^k \partial_k f, \tag{2}$$

where $\Gamma_{ij}^k$ is connection coefficients, or Christoffel's symbols of the second kind, which are essentially geodesic curvatures of the isoparametric lines that can be used to correct the errors due to the curved coordinates. The connection coefficients can be computed by

$$\Gamma_{ij}^k = \partial_i \partial_j \boldsymbol{r} \cdot \boldsymbol{G}^k, \boldsymbol{G}^k = g^{ki} \boldsymbol{g}_i, \{g^{ij}\} = \{g_{ij}\}^{-1}, g_{ij} = \boldsymbol{g}_i \cdot \boldsymbol{g}_j, \boldsymbol{g}_i = \partial_i \boldsymbol{r}, \tag{3}$$

where $\boldsymbol{r}, \boldsymbol{g}_i, \boldsymbol{g}^k$ are the position vector ($\boldsymbol{r} = (x \quad y)^T$), the basis vectors, and the dual basis vectors, respectively.

*3.2.2. Fully conjugate alignment*

We also found another alignment condition, which is represented as

$$h\{\bar{v}^i \bar{v}^j\}H = \mathrm{sym}, \tag{4}$$

where $\bar{v}^i$ is an arbitrarily prescribed vector. We assume it is normalized on the $x - y$ plane. By imposing this condition, one of the two directions in the two conjugate nets is aligned with $\bar{v}^i$. Although the second direction is also aligned between the two nets, we have no control over this direction.

### *3.2.3. Approximate bidirectional grid alignment*

It is reasonable to let the user of the method sketch a bidirectional reference grid. Assuming that two guide vectors, $\bar{v}^i$ and $\bar{s}^i$, are obtained from a reference grid, a more controlled alignment can be done by imposing

$$h\{E^{ij}\}H = \mathrm{sym}, \quad E^{ij} = \bar{v}^i\bar{v}^j + \epsilon\bar{s}^i\bar{s}^j. \tag{5}$$

Assuming a constant $\epsilon$ at each point, $E^{ij}$ becomes a constant matrix, $\bar{E}^{ij}$, and the alignment condition becomes strictly bilinear; this can be solved using VarPro, which was used to solve the equilibrium equation in [1].

## 4. Result 1 (solving only for $z$)

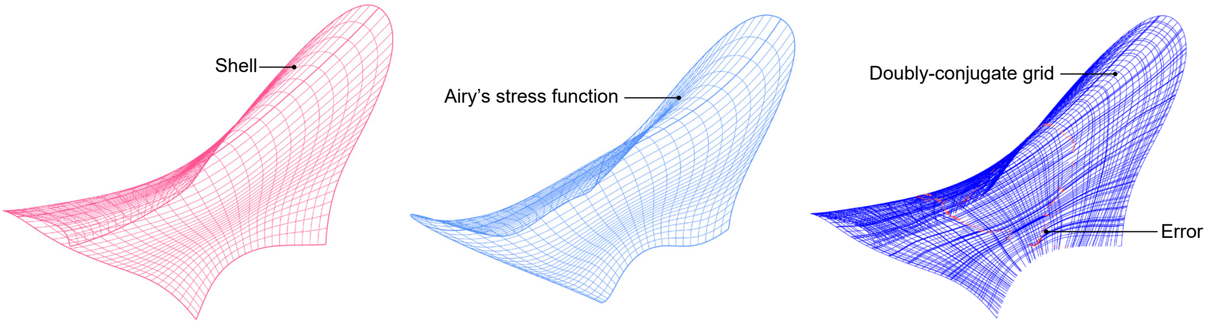

Figure 1: An example of a doubly-conjugate grid is obtained by solving the equilibrium equation and the alignment condition simultaneously using VarPro.

By imposing one of the alignment conditions, a pair of vectors simultaneously conjugate with $h$ and $H$ is guaranteed to exist. It can be easily computed by solving a generalized eigenvalue problem (see [2] and [14]). By tracing the obtained conjugate vectors, one can obtain a grid that gives a basis geometry for a bending-free grid shell and can be flat-panel-ed. We refer to this grid as a *doubly conjugate grid*. Fig.1 depicts the doubly conjugate grid obtained on the solution computed using our method. It is noteworthy that the obtained grid is visually clean and smooth. This is a significant merit in the architectural design context, although it is difficult to quantify and, thus, excluded from the mathematical discussion.

## 5. Boundary-shape optimization (experimental)

In [14], we admitted that a few example problems did not pass the post-verification method, also called the recovery test. In the recovery test, we *pin* the stress function and compute only the shell. If the stress function $\phi$ and the shell $z$ do not contradict each other, nearly the same shell should be recovered. However, different shells were recovered in a few examples. This is not surprising: the existence of a solution is not strictly guaranteed because we now have two conditions in two unknown functions (unlike the original case discussed in [1], where one condition in two unknown functions was solved and, thus, the authors were unconcerned about the existence of solutions).

Hence, in addition to the simultaneous optimization of $z$ and $\phi$ presented above, we are currently testing boundary-shape optimization—that is, solving the same equations by regarding the $x$ and $y$ coordinates of NURBS control points in addition to $z$ and $\phi$ as unknown variables. Note that the content in this section is only preliminary and experimental.

From a theoretical perspective, this boundary-shape optimization is not complex, as we only need to compute the partial derivatives of the conditions in terms of the $x$ and $y$ coordinate parameters of the

NURBS control points, just like the $z$ coordinates of the NURBS control points. However, those derivatives are in fact too complicated; thus, we explored a simplified approach.

In the original method, we solved the following conditions (excluding the continuity conditions between multiple NURBS patches and boundary conditions):

$$\begin{aligned}
(\bar{\nabla}_{ij}z)\bar{S}^{ij} &= \frac{\sqrt{g}}{\sqrt{\bar{g}}}\bar{\rho} \quad \text{where} \quad g = \mathrm{det} g_{ij} \\
\frac{1}{\sqrt{\bar{g}}}\{(\bar{\nabla}_{ij}z)\bar{E}^{jk}\bar{\nabla}_{kl}\phi\}|_{12-21} &= 0 \\
\bar{S}^{11} = \frac{1}{\bar{g}}\bar{\nabla}_{22}\phi, \bar{S}^{22} &= \frac{1}{\bar{g}}\bar{\nabla}_{11}\phi, \bar{S}^{12} = -\frac{1}{\bar{g}}\bar{\nabla}_{12}\phi, \\
\bar{\nabla}_{ij}f &= \partial_i\partial_j f - \bar{\Gamma}^k_{ij}\partial_k f,
\end{aligned} \tag{6}$$

where bar-ed symbols are quantities or operators associated with the input shape projected onto the $x - y$ plane, and $\bar{g}_{ij}$ and $\bar{\Gamma}^k_{ij}$ are the metric tensor and the connection coefficients (Christoffel's symbol of the second kind), respectively, that do not change during the computation. The naked $g$ is a metric tensor of the actual shell (solution) that considers the slope of the shell. In addition, $\phi$ and $z$ are the two unknown independent functions to be identified as a solution pair that represents the value of an Airy stress function and the height of the shell at each point. The stress tensor $\bar{S}^{ij}$ always satisfies

$$\bar{\nabla}_i\bar{S}^{ij} = 0 \quad \text{for} \quad j = 1, 2. \tag{7}$$

Because we modify the $x$ and $y$ coordinates of the NURBS control points, we deal with three different configurations: first, the input shape projected onto the $x-y$ plane; second, the deformed shape projected onto the $x - y$ plane; and third, the shape of the shell (solution). In the following account, we add a $\bar{\cdot}$ and a $\tilde{\cdot}$ to the symbols associated with the first (i.e., initial) and second (i.e., deformed) configurations, respectively. Note that "naked" symbols are measured on a shell surface with slopes. In addition, note that bar-ed symbols are independent of $x$ and $y$ and never change during the computation.

To prevent the interior points from moving freely, we consider stretching the "projected" shape by maintaining the stress tensor in the initial configuration. When a stress tensor in the initial configuration is given as $\bar{S}^{ij}$, it would be worth observing what happens when we maintain it as components of the 2nd Piola Kirchhoff (PK) stress tensor determined on the initial configuration.

The definition and behavior of the 2nd PK stress tensor on parametric surfaces are summarized in [17] (page 8). Denoting the "actual" stress tensor (known as a Cauchy stress tensor) in the deformed configuration as $\boldsymbol{\sigma} = \sigma^{ij}\boldsymbol{g}_i \otimes \boldsymbol{g}_j$, where $\boldsymbol{g}_i$ are basis vectors on the deformed configuration, the 2nd PK stress tensor is defined as

$$S = (\mathrm{det}F)F^{-1}\boldsymbol{\sigma}F^{-T}, \tag{8}$$

where $F$ is a deformation gradient tensor. A deformation gradient tensor can convert $d\boldsymbol{X}$ to $d\boldsymbol{x}$, where $d\boldsymbol{X}$ is a small vector embedded in the initial configuration and $d\boldsymbol{x}$ is the same vector after the parametric surface is stretched. Because the vector is "glued" to the surface, their components are maintained during the deformation, i.e., when $d\boldsymbol{X} = d\theta^i\boldsymbol{G}_i$, where $\boldsymbol{G}_i$ is the basis vector in the initial configuration, then $d\boldsymbol{x} = d\theta^i\boldsymbol{g}_i$ with the same $d\theta^i$, where $\boldsymbol{g}_i$ is the basis vectors in the deformed configuration. Thus, we observe that if we define the deformation gradient as $F = \boldsymbol{g}_i \otimes \boldsymbol{G}^i$, where $\boldsymbol{G}^i$ is the dual basis vectors in the initial configuration, we get $Fd\boldsymbol{X} = d\boldsymbol{x}$. Using $F^{-1} = \boldsymbol{G}_i \otimes \boldsymbol{g}^i$ and $F^{-T} = \boldsymbol{g}^i \otimes \boldsymbol{G}_i$ we get $S = S^{ij}\boldsymbol{G}_i \otimes \boldsymbol{G}_j = J\sigma^{ij}\boldsymbol{G}_i \otimes \boldsymbol{G}_j$, where $J = \sqrt{\mathrm{det}g_{ij}}/\sqrt{\mathrm{det}G_{ij}}$. Therefore, aside from the basis vectors associated with the tensors, the relation between the components with upper indices is simple:

$$\sigma^{ij} = J^{-1} S^{ij}. \tag{9}$$

Thus, returning to the $\bar{\cdot}$ / $\tilde{\cdot}$ / $\cdot$ convention, we set

$$\tilde{S}^{ij} = J^{-1} \bar{S}^{ij}, \quad \text{where} \quad J = \frac{\sqrt{\tilde{g}}}{\sqrt{\bar{g}}}. \tag{10}$$

In the following, we prove that the horizontal equilibrium of this stress tensor is satisfied when

$$(\bar{\nabla}_{ij}\tilde{x})\bar{S}^{ij} = 0, \quad \text{and} \quad (\bar{\nabla}_{ij}\tilde{y})\bar{S}^{ij} = 0, \tag{11}$$

where $(\tilde{x}, \tilde{y})$ represents a position vector of each point after the deformation. This is a beautiful analogy of the equilibrium equation in $z$ (the first equation in Eq. (6), known as Pucher's equation). Please also note that these conditions are strictly bilinear and, thus, can be solved using VarPro.

The horizontal equilibrium in the deformed configuration is

$$\tilde{\nabla}_i \tilde{S}^{ij} = 0, \tag{12}$$

but the horizontal equilibrium guaranteed to be satisfied is Eq. (7). Hence, we seek a condition to fulfill

$$\tilde{\nabla}_i \tilde{S}^{ij} - (\bar{\nabla}_i \bar{S}^{ij}) J^{-1} = 0. \tag{13}$$

We begin with

$$\begin{aligned} (\bar{\nabla}_i \bar{S}^{ij}) J^{-1} &= J^{-1}(\partial_i \bar{S}^{ij} + \bar{\Gamma}^j_{im} \bar{S}^{im} + \bar{\Gamma}^i_{im} \bar{S}^{mj}) \\ \tilde{\nabla}_i \tilde{S}^{ij} &= J^{-1}(\partial_i \bar{S}^{ij} + \tilde{\Gamma}^j_{im} \bar{S}^{im} + \tilde{\Gamma}^i_{im} \bar{S}^{mj}) + (\tilde{\nabla}_i J^{-1}) \bar{S}^{ij}. \end{aligned} \tag{14}$$

But, using $D\sqrt{\det g_{ij}} = 1/2(Dg_{\alpha\beta})g^{\alpha\beta}\sqrt{\det g_{ij}}$, where $D$ is a generic differential operator, $\tilde{\nabla}_i J^{-1}$ expands as

$$\begin{aligned} \tilde{\nabla}_i J^{-1} &= \frac{1}{2} J^{-1}((\tilde{\nabla}_i \bar{g}_{\alpha\beta})\bar{g}^{\alpha\beta} - (\tilde{\nabla}_i \tilde{g}_{\alpha\beta})\tilde{g}^{\alpha\beta}) \\ &= \frac{1}{2} J^{-1}((\tilde{\nabla}_i \bar{g}_{\alpha\beta})\bar{g}^{\alpha\beta}) \quad \because \tilde{\nabla}_i \tilde{g}_{\alpha\beta} = 0. \end{aligned} \tag{15}$$

This further simplifies as

$$\begin{aligned} \frac{1}{2} J^{-1}(\tilde{\nabla}_i \bar{g}_{\alpha\beta})\bar{g}^{\alpha\beta} &= \frac{1}{2} J^{-1}(\partial_i \bar{g}_{\alpha\beta} - \tilde{\Gamma}^m_{i\alpha}\bar{g}_{m\beta} - \tilde{\Gamma}^m_{i\beta}\bar{g}_{\alpha m})\bar{g}^{\alpha\beta} \\ &= J^{-1}(\bar{\Gamma}^m_{mi} - \tilde{\Gamma}^m_{mi}) \quad \because (\partial_i \bar{g}_{\alpha\beta})\bar{g}^{\alpha\beta} = 2\bar{\Gamma}^m_{im}. \end{aligned} \tag{16}$$

Therefore, we obtain

$$\tilde{\nabla}_i \tilde{S}^{ij} - (\bar{\nabla}_i \bar{S}^{ij}) J^{-1} = J^{-1}(\tilde{\Gamma}^j_{\alpha\beta} - \bar{\Gamma}^j_{\alpha\beta})\bar{S}^{\alpha\beta}. \tag{17}$$

Thus, having $J \neq 0$ and $\bar{\nabla}_i \bar{S}^{ij} = 0$, we observe that the horizontal equilibrium in the deformed configuration is satisfied by fulfilling

$$(\tilde{\Gamma}^k_{ij} - \bar{\Gamma}^k_{ij})\bar{S}^{ij} = 0 \quad \text{for} \quad k = 1, 2. \tag{18}$$

This condition further simplifies to Eq. (11). Note that the initial position vector $\bar{\boldsymbol{r}} = (\bar{x} \quad \bar{y})$ satisfies this horizontal equilibrium.

When the horizontal equilibrium condition is imposed, the interior points do not move freely. However, because the VarPro method is basically a least squares minimization method, the projected shape overly shrinks until the area becomes nearly zero. Scaling the stress tensor with $J^{-1}$ is effective in canceling out this scaling issue.

Following the above procedure, we can keep using numerous quantities associated with the initial configuration instead of solving the equations on the deformed configuration. Note that, for the alignment conditions, we must use $\tilde{\nabla}_{ij}z$ instead of $\bar{\nabla}_{ij}z$ because we are aligning the "actual" curvature with the stress tensor. Thus, the equations to be solved are

$$
\begin{aligned}
J^{-1}(\bar{\nabla}_{ij}z)\bar{S}^{ij} &= \frac{\sqrt{g}}{\sqrt{\tilde{g}}}\bar{\rho} \\
\frac{1}{\sqrt{\bar{g}}}\{(\tilde{\nabla}_{ij}z)\bar{E}^{jk}\bar{\nabla}_{kl}\phi\}|_{12-21} &= 0 \\
\bar{S}^{11} = \frac{1}{\bar{g}}\bar{\nabla}_{22}\phi, \bar{S}^{22} &= \frac{1}{\bar{g}}\bar{\nabla}_{11}\phi, \bar{S}^{12} = -\frac{1}{\bar{g}}\bar{\nabla}_{12}\phi, \\
\bar{\nabla}_{ij}f &= \partial_i\partial_j f - \bar{\Gamma}^k_{ij}\partial_k f, \\
\tilde{\nabla}_{ij}f &= \partial_i\partial_j f - \tilde{\Gamma}^k_{ij}\partial_k f, \\
J^{-1}(\bar{\nabla}_{ij}x)\bar{S}^{ij} &= 0, \\
J^{-1}(\bar{\nabla}_{ij}y)\bar{S}^{ij} &= 0.
\end{aligned}
\tag{19}
$$

Among these equations, the most difficult factor in calculating the gradients is $\tilde{\Gamma}^k_{ij}$. We do not show the proof here, but one can rely on the following relation:

$$
D\tilde{\Gamma}^k_{ij} = (\tilde{\nabla}_{ij}Dx)(\tilde{\boldsymbol{G}}^k \cdot \boldsymbol{e}_x) + (\tilde{\nabla}_{ij}Dy)(\tilde{\boldsymbol{G}}^k \cdot \boldsymbol{e}_y), \tag{20}
$$

where $D$ is a generic differential operator.

In addition to these equations, a good boundary fairness/elasticity term that does not interfere with the horizontal equilibrium is required. To this end, we are currently testing the following conditions on free edges:

$$
(\bar{\nabla}_{ij}x)\bar{v}^i\bar{v}^j = 0, \quad (\bar{\nabla}_{ij}y)\bar{v}^i\bar{v}^j = 0, \tag{21}
$$

where $\bar{v}^i$ is a normalized vector perpendicular to the edge. On a free edge, the stress tensor contains only a component parallel to the edge, i.e., $\bar{S}^{ij} = \lambda\bar{v}^i\bar{v}^j$. Hence, these conditions are already contained in the horizontal equilibrium and thus do not interfere with it. Because the horizontal equilibrium is equivalent to Eq. (18), these conditions are further equivalent to

$$
(\tilde{\Gamma}^k_{ij} - \bar{\Gamma}^k_{ij})\bar{v}^i\bar{v}^j = 0 \quad \text{for} \quad k = 1, 2. \tag{22}
$$

Hence, these conditions work as an aesthetic/elastic term that can preserve some curvature of the free edges.

## 6. Result 2 (solving for $x$, $y$, and $z$)

Fig. 2 depicts shells, Airy stress functions, results of the recovery test, and doubly conjugate grids computed using the above-described strategy. Comparisons of doubly conjugate grids obtained with

Airy's stress function and an FEA analysis result are also provided. Between (a) a solution obtained by optimizing $z$ only and (b) a solution obtained by optimizing $x$, $y$, and $z$, the quality of the recovery test (how well the recovered surface overlays with the solution) was improved in (b), and the number of red dots, which represent locations where the computation of a doubly conjugate grid is difficult, were significantly reduced as well. Moreover, the grid obtained with an FEA analysis result overlays well with the "theoretical" grid computed with the obtained Airy's stress function. We will continue this research in this direction.

## 7. Conclusion

We derived a new bilinear symmetric PDE that can be used to align conjugate stress and curvature nets. This PDE has the same form as the equilibrium equation. Fortunately, we have already reported that the equilibrium equation, which is also a bilinear PDE, can be solved using VarPro in 2022[1]. In this context, we demonstrated that the two bilinear PDEs can be simultaneously solved using VarPro. Remarkably, not only are the two nets aligned, but the resulting grid is also visually clean and appealing. This technique could enable us to design metal–glass grid shells that are bending-free and can be simultaneously covered by planar quadrilateral glass panels.

Apart from introducing the alignment conditions, we presented an additional result in which boundary shape optimization was experimentally activated. The aesthetic quality of the obtained grids was improved, and the number of errors in them was drastically reduced. Note that this feature is only experimental, and we are currently running verifications on numerous example problems.

Our sample implementation of the method explained in this paper, which runs on Rhinoceros ®, is available on Food4Rhino (https://www.food4rhino.com/en/resource/godzilla-v10).

## Acknowledgments

This research was partially supported by the Nohmura Foundation, the Maeda Engineering Foundation, and JSPS KAKENHI Grant Number 23K17784.

## References

[1] M. Miki and T. Mitchell, "Interactive exploration of tension-compression mixed shells," *ACM Transactions on Graphics (TOG)*, vol. 41, no. 6, Nov. 2022, ISSN: 0730-0301.

[2] E. Vouga, M. Höbinger, J. Wallner, and H. Pottmann, "Design of self-supporting surfaces," *ACM Transactions on Graphics (TOG)*, vol. 31, no. 4, pp. 1–11, Jul. 2012.

[3] M. R. Barnes, "Form finding and analysis of tension structures by dynamic relaxation," *International journal of space structures*, vol. 14, no. 2, pp. 89–104, 1999.

[4] H.-J. Schek, "The force density method for form finding and computation of general networks," *Computer methods in applied mechanics and engineering*, vol. 3, no. 1, pp. 115–134, 1974.

[5] A. Kilian and J. Ochsendorf, "Particle-spring systems for structural form finding," *Journal of the international association for shell and spatial structures*, vol. 46, no. 2, pp. 77–84, 2005.

[6] S. Adriaenssens, L. Ney, E. Bodarwe, and C. WILLIAMS, "Dutch maritime museum: Form-finding of an irregular faceted skeletal shell-part b," in *Symposium of the International Association for Shell and Spatial Structures (50th. 2009. Valencia). Evolution and Trends in Design, Analysis and Construction of Shell and Spatial Structures: Proceedings*, Editorial Universitat Politècnica de València, 2010.

[7] D. Piker, "Kangaroo: Form finding with computational physics," *Architectural Design*, vol. 83, no. 2, pp. 136–137, 2013.

[8] A. E. Green and W. Zerna, *Theoretical elasticity*. London: Oxford University Press, 1968.

[9] P. Csonka, *Theory and practice of membrane shells*. Düsseldorf, Germany: VDI Verlag, 1987.

[10] E. Sanchez-Palencia, O. Millet, and F. Béchet, *Singular problems in shell theory: computing and asymptotics. Lecture Notes in Applied and Computational Mechanics.* Berlin: Springer-Verlag, 2010, vol. 54.

[11] Y.-C. Chiang and A. Borgart, "A form-finding method for membrane shells with radial basis functions," *Engineering Structures*, vol. 251, p. 113 514, 2022.

[12] G. H. Golub and V. Pereyra, "The differentiation of pseudo-inverses and nonlinear least squares problems whose variables separate," *SIAM Journal on numerical analysis*, vol. 10, no. 2, pp. 413–432, 1973.

[13] L. Eldén and S. Ahmadi-Asl, "Solving bilinear tensor least squares problems and application to hammerstein identification," *Numerical Linear Algebra with Applications*, vol. 26, no. 2, e2226, 2019.

[14] M. Miki and T. Mitchell, "Alignment conditions for nurbs-based design of mixed tension-compression grid shells (to appear)," *ACM Transactions on Graphics (TOG)*, vol. 43, no. 4, Jul. 2024, ISSN: 0730-0301.

[15] D. Pellis and H. Pottmann, "Aligning principal stress and curvature directions.," in *AAG 2018*, 2018, pp. 34–53.

[16] M. Kilian, D. Pellis, J. Wallner, and H. Pottmann, "Material-minimizing forms and structures," *ACM Trans. Graph.*, vol. 36, no. 6, Nov. 2017, ISSN: 0730-0301.

[17] K.-U. Bletzinger, *Form finding and optimization of membranes and minimal surfaces*. Institut für Baustatik Stuttgart, Germany, 1998.

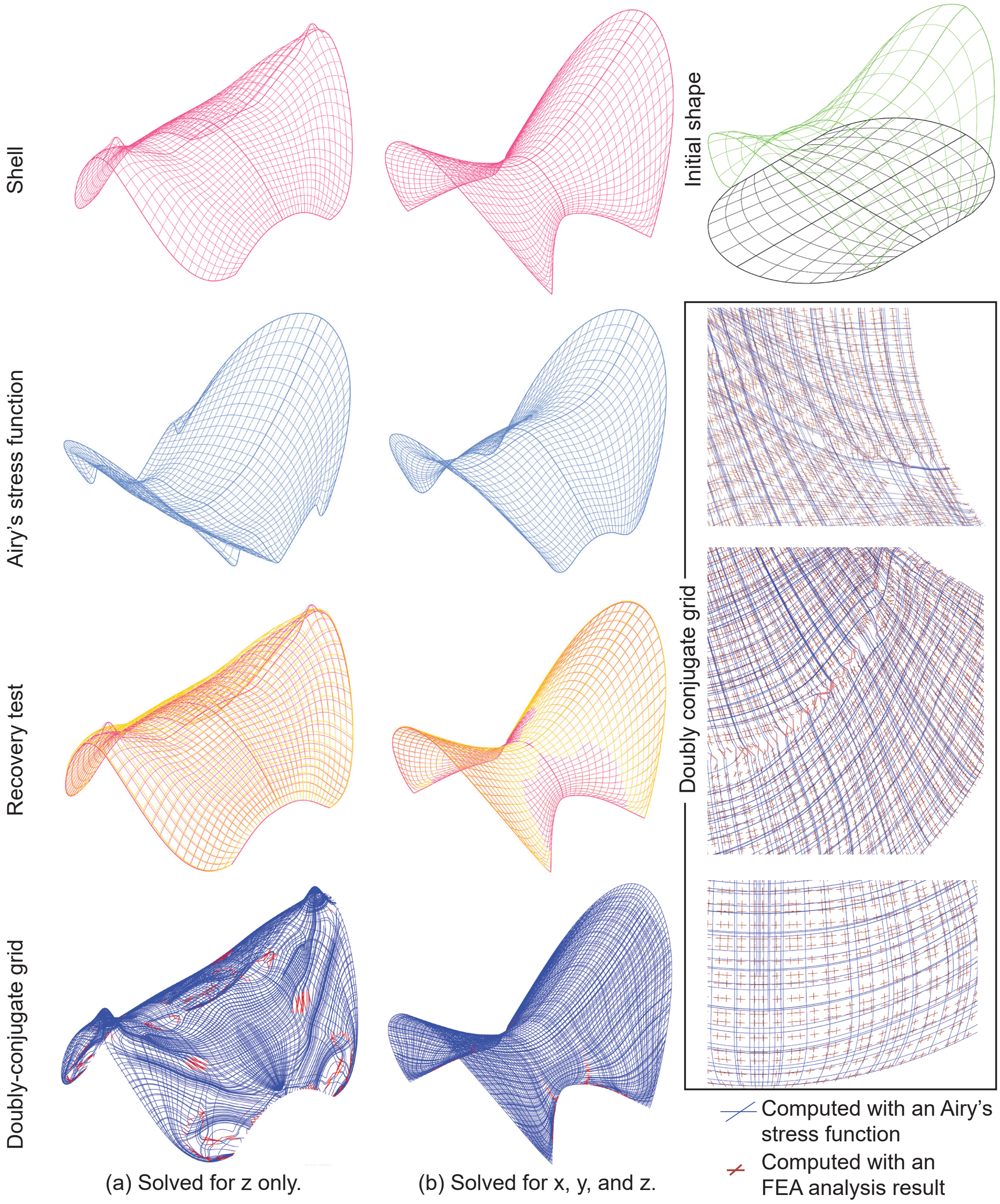

Figure 2: Form-finding results of the STARFISH example.