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.)

Coupling form-finding methods for efficient structural shape optimization via gradient descent

Rafael PASTRANA[*,a], Sigrid ADRIAENSSENS[a]

[*] School of Architecture, Princeton University
Princeton, NJ, United States of America
arpastrana@princeton.edu

[a] Form Finding Lab, Department of Civil and Environmental Engineering, Princeton University

Abstract

Form-finding methods enable the design of lightweight structures like thin shells and long-span bridges. While form-finding methods share the goal of computing shapes in static equilibrium, each method is built around specific numerical assumptions that make them specialize to a particular group of structural typologies. However, practical lightweight structures are often assemblies of distinct substructures and typologies, conditioned on specific geometric, fabrication, and mechanical constraints. While approaches to solving such constrained form-finding problems exist, they show limited success in computing optimal shapes in complex design tasks. In this paper, we explore an integrated form-finding method that couples the force density method (FDM) and the combinatorial equilibrium modeling (CEM) framework using automatic differentiation and gradient-based optimization. Our approach facilitates direct communication between the FDM and the CEM, enabling efficient shape optimization of complex lightweight structures. In a case study, we demonstrate that our approach computes equilibrium shapes that meet target constraints with ten times higher accuracy than current methods, and without extra computational cost. By leveraging the individual strengths of different form-finding methods in a unified workflow, we foster the development of enhanced optimization tools for structural design.

Keywords: Lightweight structures, form-finding, structural optimization, automatic differentiation, constraints

1 Introduction

Form-finding methods facilitate the conceptual design of lightweight structures as they generate shapes that enable these structures to withstand loads predominantly via axial forces. As a result of this mechanical behavior, a structure bears loads with less material volume than a structure whose shape has not been form-found. Several geometry-based form-finding methods have been developed over the last fifty years to design lightweight structures modeled as pin-jointed, bar frameworks [1]. The array of methods is wide, ranging from the canonical force density method (FDM) [2] originated in the 1970s, to contemporary approaches such as the thrust network analysis (TNA) [3] and the combinatorial equilibrium modeling (CEM) framework [4] invented in the 2000s.

While form-finding methods have the same general objective, which is to compute static equilibrium shapes, they do so by making specific numerical and modeling assumptions. These assumptions specialize the method, explicitly or not, to a particular set of structural typologies. For example, the TNA, frequently applied in thin masonry shell design, fixes a structure's projection on-plan for vertical loads by separating horizontal and vertical equilibrium calculations [5, 6]. The CEM caters to the optimization of structures like bridges and towers because it propagates static equilibrium sequentially in a layered structure [7, 8]. In contrast, the FDM computes vertical and horizontal equilibrium configurations at

Copyright ©2024 by Rafael PASTRANA, Sigrid ADRIAENSSENS.
Published in the Proceedings of the IASS Annual Symposium 2024 with permission.

Figure 1: Lightweight structures are assemblies of distinct substructures and typologies. Left: a cable-stayed bridge. Middle: a post-tensioned observation tower. Right: a membrane shading system.

the same time on unstructured networks, and is commonly used for cablenet, membrane, and tensegrity optimization [9, 10].

Nevertheless, real-world lightweight structures are complex because they often are assemblies of different shells, trusses, cablenets, and other structural typologies. Moreover, each of them is subject to particular constraints that arise, for instance, from architectural and fabrication requirements in a design project. Despite their apparent discrepancies, these structural components must work together to resist applied loads and transfer them towards the structure's foundations.

To address these constrained form-finding problems, it is often resorted to two optimization-based approaches. The first approach optimizes all structural components simultaneously with a single form-finding method. This would amount to optimizing the shapes of the deck and cables of the bridge displayed in Fig. 1 at the same time. But this often results in poor convergence due to conflicting constraints, leading to unsatisfactory solutions. The second approach, which aligns more with engineering intuition, involves form-finding each component independently, passing loads from one component to the next. For instance, in the observation tower in Fig. 1, the external cables can be optimized first, followed by the observation platforms, and the interior masts at last. The drawback of the second approach is that optimizing components independently can hinder subsequent components in the chain from reaching an optimal shape. In general, both approaches require potentially extensive calibration to produce adequate geometries, slowing the design process and limiting the integration of geometric form-finding methods into current, digital structural design workflows.

How to combine form-finding methods more efficiently to compute optimal shapes for lightweight structural design? This paper explores a third approach by coupling two different form-finding methods in a unified computation pipeline via automatic differentiation and gradient-based optimization (Section 2). To this end, we study the effect of having the FDM and the CEM communicate during form-finding and optimization in the search for an optimal shape in equilibrium for a cablenet-stayed bridge that fulfills a priori design constraints (Section 3). The proposed coupling enables the simultaneous optimization of the parameters of both form-finding methods to minimize a loss function end-to-end. Our experiments show that our approach outperforms other existing methods in generating optimal shapes for complex lightweight structures. The paper ends in Section 4 where we provide conclusions and outline future work directions.

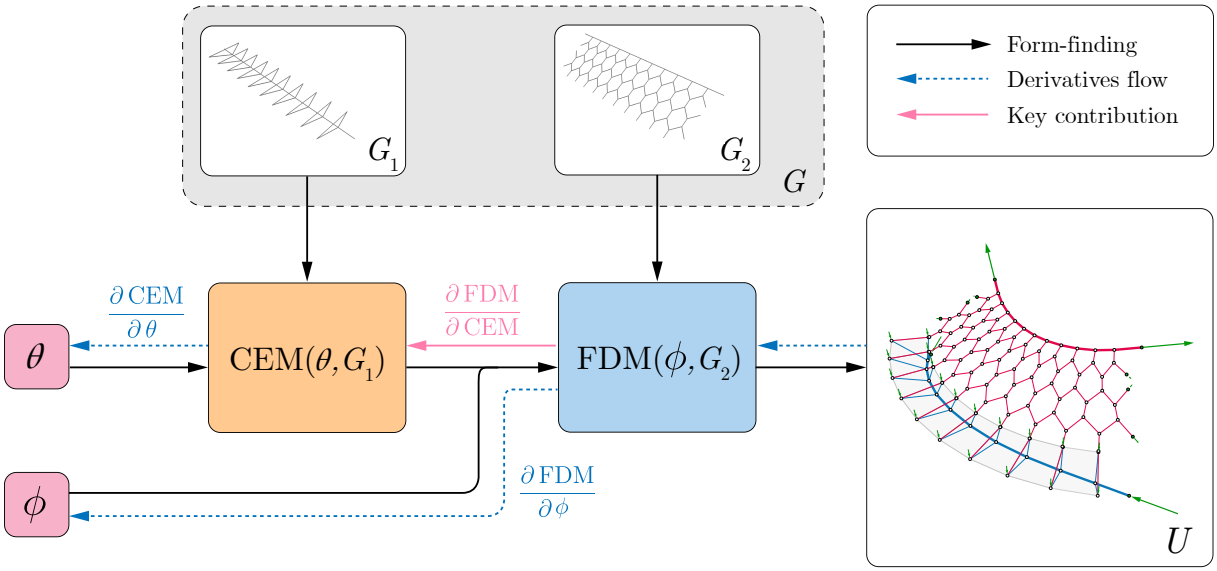


Figure 2: We integrate the CEM framework and the FDM in a single computation pipeline to efficiently optimize the structural geometry of a cablenet-stayed bridge.

2 Methods

Our goal is to calculate optimal equilibrium shapes by coupling two form-finding methods with the approach illustrated in Fig. 2. Section 2.1 explains how to compute an equilibrium shape with the FDM and the CEM, first separately and then jointly. Afterwards, Section 2.2 describes how to constrain the calculation of that shape by feeding the form-finding methods' derivatives to a gradient-based optimizer.

2.1 Equilibrium computation

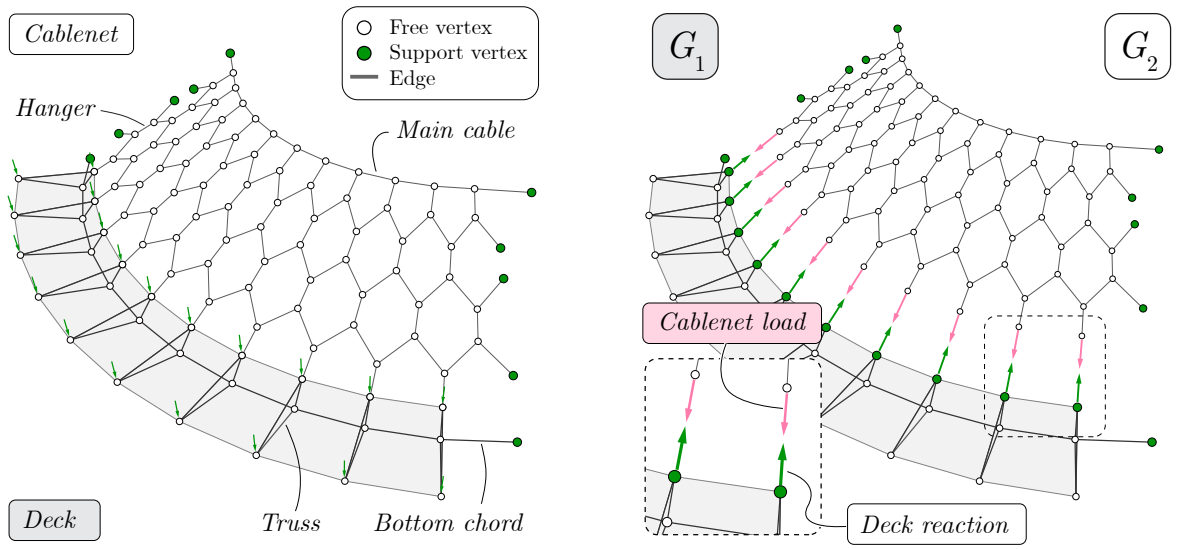
In our approach, a pin-jointed bar structure is modeled as graph G, where the graph edges correspond to the structure's bars, and its vertices correspond to the structure's free and supported joints (Fig. 3a). A form-finding method then calculates a state of static equilibrium U given specific load and support conditions. The primary attribute in U is the position in equilibrium of every unsupported vertex, $\mathbf{x}$ (i.e. the shape in static equilibrium of the structure). The other attributes that complete U are the bar axial forces f (tension or compression) and lengths l, and the reaction forces at the supports, $\mathbf{r}$. They are calculated after generating a shape in equilibrium. In addition to G, a form-finding method requires a set of continuous parameters as input to calculate U. These parameters are specific to the FDM and the CEM way of computing equilibrium.

2.1.1 Force density method (FDM)

The FDM requires parameters ϕ to compute a state of static equilibrium U on G:

$$U = \text{FDM}(\phi, G)$$

These parameters are the force densities of the graph edges (which define the ratios between the bar forces and lengths in the equilibrium state), the applied loads, and the position of the support vertices [2].



(a) Structural system on graph G

(b) Mechanical coupling between graphs G_1, G_2

Figure 3: We model a cablenet-stayed bridge as a single graph G and apply loads and support conditions to create a structural system for form-finding. We can alternatively model this structure as a pair of graphs G_1 and G_2 mechanically coupled across a shared interface of coincident vertices.

To obtain the free vertices position in static equilibrium $\mathbf{X}$, the FDM solves a linear system of equations:

$$\mathbf{X} = \mathbf{K}^{-1}\,\mathbf{P}$$

In this system, $\mathbf{K}$ is the geometric stiffness matrix given by the bars' force density and the graph's connectivity and $\mathbf{P}$ corresponds to the loads applied to the free vertices, minus the contribution of the fixed vertices forces. Additional details on calculating the other components of U are found in [2, 11].

2.1.2 Combinatorial equilibrium modeling (CEM)

The CEM also operates on an input graph G, but unlike the FDM, the edges of G must be labeled as either trail or deviation edges [4]. One of the benefits of labeling the edges is that it allows the CEM to directly accept forces f and lengths l, in addition to the applied loads, as the input parameters θ for form-finding:

$$U = \text{CEM}(\theta, G)$$

Furthermore, the distribution of edge labels dictates how to group the vertices of G into a sequence of vertex layers. The order of these layers is relevant because the CEM computes the position in equilibrium of the free vertices one layer at a time. More specifically, the position $\mathbf{X}_{k+1}$ of the vertices in the $k+1$ layer depends on the position $\mathbf{X}_k$ of the vertices in the previous layer, k:

$$\mathbf{X}_{k+1} = \mathbf{X}_k + \mathbf{l}_k^{\mathrm{T}}\,\tilde{\mathbf{r}}_k$$

where $\mathbf{l}_k$ represents the signed lengths of the trail edges connecting layers k and $k+1$; and $\tilde{\mathbf{r}}_k$, is the unit-length residual vector resulting from adding the contribution of the force vectors of the deviation edges and the applied loads incident to k. The remainder of U can be computed with the formulae given in [4, 7].

2.1.3 Coupled equilibrium

Now, we look at how to generate an equilibrium state U with the FDM and the CEM while maintaining mechanical compatibility.

First, we split the graph G that represents the structure of interest into subgraphs G_1 and G_2. The partitioning is arbitrary, but it should ideally be done by breaking down the structure into distinct components following the load path. In Fig. 3, for example, we split a stayed bridge into a deck and a cablenet. After splitting, we add temporary supports to the group of vertices of G_1 at the interface with G_2 (Fig. 3b).

Then, we perform the computation of an equilibrium state for G in two stages. In the first stage, we calculate an equilibrium substate U_1 on G_1 with one form-finding method, and in the second stage, a substate U_2 on G_2 with another method. Here, we employ the CEM and the FDM for the first and second stages because of their documented suitability to form-find bridges and cable structures, respectively [10, 4, 7]. The equilibrium state U of the global structure results from concatenating both substates.

One of the key ideas in our approach is that the equilibrium state generated by the CEM supplies a portion of the input parameters of the FDM:

$$U = \text{FDM}(\text{CEM}(\theta, G_1), \phi, G_2)$$

In Fig. 3b, the data transferred from the CEM to the FDM are the reaction forces on the temporarily supported side of the deck. However, parameter transfer alone is insufficient to ensure mechanical compatibility at the interface between graphs. To achieve mechanical coupling, the reaction forces are applied as loads to the cablenet vertices that interface with the deck, and the position of these cablenet vertices must match that of their counterparts on the deck. While imposing the first condition is straightforward, fulfilling the second one needs optimization.

2.2 Constrained form-finding with gradient descent

Classic form-finding methods have to be complemented by other numerical techniques to constrain an equilibrium state so that it satisfies additional geometric, fabrication, and mechanical requirements. Here, we cast such a constrained form-finding task as an optimization problem following a penalty approach, by formulating a loss function $\mathcal{L}(\theta, \phi)$ in terms of the FDM and the CEM parameters:

$$\mathcal{L}(\theta, \phi) = ||\hat{U} - U(\theta, \phi)||^2$$

The loss function measures the squared distance between the attributes of an equilibrium state U produced by the coupled CEM and FDM (Section 2.1), and those in a target state $\hat{U}$ with the constraints. Next, we solve the optimization problem by finding the parameter values that minimize the loss. By reducing the distance between states, the resulting optimal parameters produce an equilibrium state that fulfills the targets in a least-squares sense:

$$\theta^\star, \phi^\star = \underset{\theta,\phi}{\text{argmin}}\, \mathcal{L}$$

We employ gradient descent and associated gradient-based optimization algorithms to minimize the loss. Gradient descent is an efficient approach to solving optimization problems by iteratively updating the continuous form-finding parameters θ, ϕ in the negative direction of the gradient of the loss:

$$\nabla_{\theta,\phi}\mathcal{L} = \frac{\partial \mathcal{L}}{\partial \text{FDM}} \left[\frac{\partial \text{FDM}}{\partial \text{CEM}} \frac{\partial \text{CEM}}{\partial \theta}, \frac{\partial \text{FDM}}{\partial \phi} \right]$$

The gradient expression provides insight into how both form-finding methods communicate during optimization: the derivative of the CEM w.r.t. its inputs θ is scaled by the derivative of the FDM output w.r.t. the CEM's output. As a result, the CEM parameters can be tuned so that the equilibrium state computed by the FDM diminishes the loss.

Traditionally, evaluating the gradient for optimization requires the formulation of analytical equations for the derivatives. This process can be time-consuming and error-prone. Instead, we conveniently obtain exact gradient values (exact up to computer precision) without analytical equations with reverse-mode automatic differentiation [12]. This is possible because we have access to auto-differentiable implementations of the CEM [7] and the FDM [11].

3 Case study

We want to understand the effect of coupling two form-finding methods on the solution of a complex form-finding task. To this end, we benchmark our method against two other current approaches [7, 11] to calculate an optimal shape for the cablenet-stayed bridge displayed in Fig. 3.

The three tested approaches are:

- **FDM:** Optimizing the entire bridge graph G using a single form-finding method, the FDM.
- **CEM/FDM:** Solving the task in two separate stages, as per Section 2.1.3, but without derivatives transfer. We optimize the deck with the CEM, fix the found parameters, and use them as static inputs to optimize the cablenet with the FDM (Fig. 3b).
- **CEM+FDM:** Our integrative approach, transferring parameters and derivatives between the two form-finding methods to solve the optimization problem end-to-end.

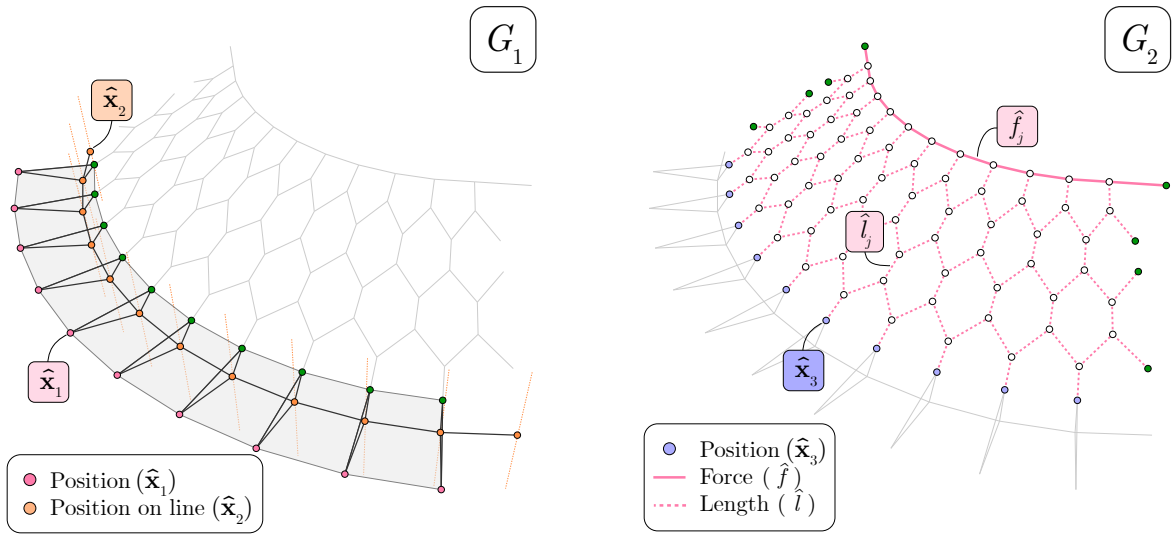
We solve all the optimization problems with L-BFGS-B [13], setting the maximum number of iterations to 500 and the convergence tolerance to 1×10^{-6}. We pick the cablenet force densities and the deck internal forces as the optimization parameters, ϕ and θ, of the FDM and the CEM methods, respectively. All three approaches are initialized with the same starting parameter values. We use JAX [14, 15, 7, 11] as the differentiable programming ecosystem to execute the experiments.

3.1 Structural system

The bridge spans 22 meters between supports, as shown in Fig. 3a. It consists of a rigid deck that is curved on-plan and suspended longitudinally on one side by a tensile cablenet. On the other side, the deck cantilevers.

The deck is initially 0.6 meters deep and 2.5 meters wide. The deck is composed of 10 triangular trusses. Every vertex at the top of the trusses is subject to a vertical point load of $5/8$ kN. The trusses transfer one portion of the applied loads to a bottom chord that runs continuously until it reaches the supports at the two support abutments of the structure. The other portion of the loads is transferred to the cablenet.

The cablenet is a network of cable hangers weaved in a hexagonal pattern, transferring their share of the deck loads progressively until they reach a main longitudinal cable at the top. The height between the top of the deck and the main cable supports is 3 meters. On plan, the cable supports are offset 5 meters from the deck.



(a) Deck (b) Cablenet

Figure 4: Task definition for the constrained form-finding of a cablenet-stayed bridge. The goals are defined on the vertices and edges of graphs G_1 and G_2.

3.2 Design constraints

The main objective of this task is to compute optimized shapes for the deck and the cablenet so that the applied loads only exert internal axial stresses in the structure. This condition is fulfilled if the entire bridge is in static equilibrium. Additionally, the deck and the cablenet shapes must comply with five constraints to produce a feasible bridge design. These requirements, illustrated in Fig. 4, are a mix of geometric, fabrication, and structural specifications:

1. **Curved deck shape:** The cantilevered side of the deck should be equally spaced on a quarter of a circle with a 15-meter radius. Therefore, the position of the deck vertices on that side ought to match the target positions $\hat{\mathbf{x}}_1$.

2. **Uniform truss spacing:** The trusses should be spaced evenly on the bottom chord. We constraint the position $\hat{\mathbf{x}}_2$ of the chord vertices to slide along a line on the same tangent plane as $\hat{\mathbf{x}}_1$.

3. **Mechanical compatibility**: The vertices at the free ends of the cablenet must reach the position $\hat{\mathbf{x}}_3$ of the vertices they interface with on the deck.

4. **Target cable lengths**: To simplify fabrication, we look for an equilibrium state where the length of the cablenet hangers is $\hat{l} = 1$ meter.

5. **Equalized cable forces**: The internal force of the edges in the main longitudinal cable should reach a value of $\hat{f} = 10$ kN to manufacture it from a single cable diameter that is fully stressed.

3.3 Results and discussion

Fig. 5 shows the convergence plots of the three evaluated approaches after solving the constrained form-finding problem.

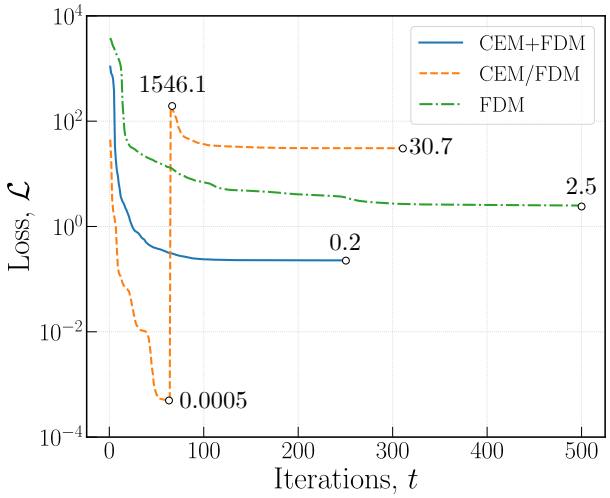


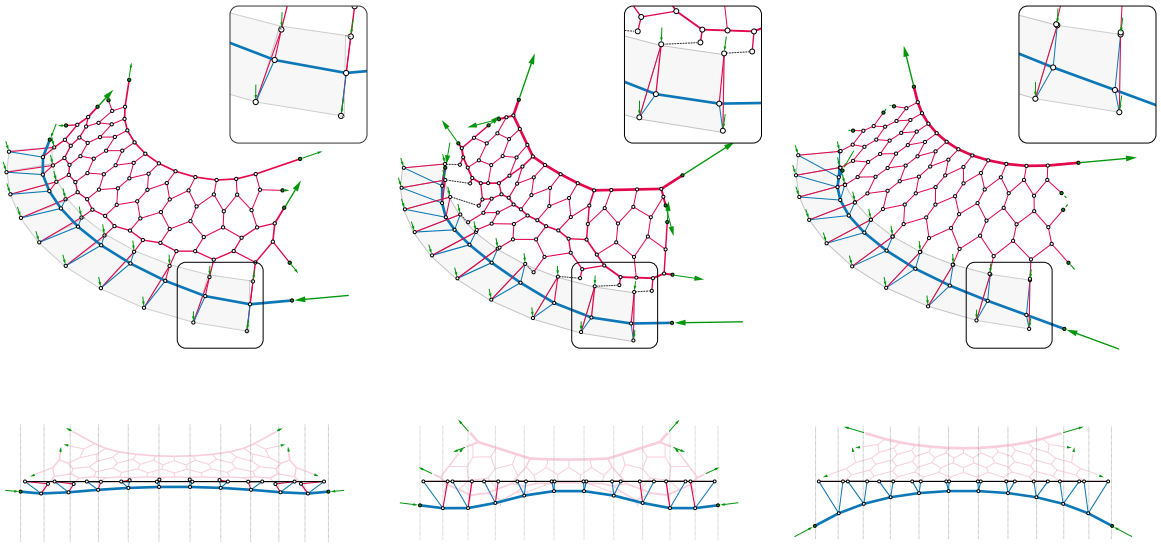
Figure 5: Convergence plots for the constrained form-finding problem solved with three different approaches to optimize the structural shape of a cablenet-stayed bridge.

The FDM optimization reaches a minimum loss value of $\mathcal{L} = 2.5$ after exhausting the maximum number of iterations in 1 second. Fig. 6a depicts the deck and cablenet's equilibrium states generated by the FDM. The deck shape exhibits adequate performance in achieving the first two constraints: the topmost deck vertices lie on the target curve, and the chord vertices stay on the target lines within a mean distance of 0.1 meters. The location of the bottom chord at the abutments is unchanged, although the chord bulges upwards at midspan. In the cablenet, the average force in the main longitudinal cable is $f = 10$ and fulfills constraint 5, but the average length of the cablenet hangers is $l = 0.91$, about 10% off from constraint 4's specification. The resulting cablenet design is thus unsatisfactory.

Next, we look at the results produced after solving the constrained form-finding problem with the CEM and the FDM separately (CEM/FDM). This disjoint optimization takes a total of 311 iterations and 2.5 seconds. The convergence curve of the CEM/FDM approach in Fig. 5 consists of two segments connected by a sharp discontinuity. The first segment reflects the convergence history of the optimization problem solved with the CEM, and the second to the one addressed with the FDM. The CEM optimization converges to a final loss value of 0.0005 in only 64 iterations, indicating that the computed deck shape fulfills constraints 1 and 2 by tightly matching the target positions. Like in the first approach, the bridge's bottom chord bulges upwards in the middle, but here the deck section is deeper (Fig. 6b).

Nevertheless, the success of the deck optimization with the CEM is at the expense of that of the cablenet with the FDM. This issue is reflected in the second segment of the CEM/FDM loss curve in Fig. 5. At the outset of the FDM optimization, the loss surges drastically from 0.0005 to 1546.1. While the gradient-based optimizer can drive the loss down fifty-fold to $\mathcal{L} = 30.7$ in 247 iterations, the combined loss value of the deck and the cablenet optimizations is the largest relative to the other two approaches (30.7 vs. 2.5 and 30.7 vs. 0.2). These results situate the CEM/FDM as the worst-performing approach for the bridge design. The cablenet shape in Fig. 6b further evidences the poor performance of optimizing the CEM and the FDM parameters separately. In particular, we note that the cablenet is unable to match the position of the deck vertices at the interface (they are on average 1.35 meters away from their target position, failing to fulfill constraint 3) and that the lengths of the hangers are visibly uneven (not meeting constraint 4).

We attribute the poor performance of the disjoint CEM/FDM approach to the lack of parameters and derivatives transfer between optimization stages. Since the first stage is unaware of the second one



(a) FDM (b) CEM/FDM (c) CEM+FDM (Ours)

Figure 6: Optimal equilibrium states for a cablenet-stayed bridge. Our approach produces the solution that best satisfies the imposed design constraints. Edges colored in blue denote axial compression, and those in red, axial tension.

during optimization, the deck shape converges to a local optimum that fulfills constraints 1 and 2, but whose resulting reaction forces at the interface, which are then applied as loads on the cablenet, then jeopardize the computation of a suitable tension-only equilibrium state in the second stage.

In contrast, the integrative CEM+FDM approach can tune the transferred reaction forces during optimization to solve the posed constrained form-finding problem successfully. Our approach monotonically converges to a final loss value of $\mathcal{L} = 0.2$ in 248 iterations (Fig. 5). This loss is at least one order of magnitude lower than the FDM approach, and two orders of magnitude lower than the CEM/FDM approach, demonstrating the advantages of our approach in calculating a shape that satisfies the 5 design constraints. Additionally, we note that good convergence does not incur additional computational cost: the optimization runtime of CEM+FDM is 1 second, equal to the FDM's, and 2.5 times faster than the disjoint CEM/FDM approach.

Visual inspection of Fig. 6c confirms the enhanced performance of our method. For example, the vertices at the interface between the deck and cablenet overlap (as the distances between them are only 0.01 meters on average); the interior cable length reaches a mean value of $l = 1$ (the target length is $\hat{l} = 1$ in constraint 4); and the forces in the main cable match the target of $\hat{f} = 10$. An interesting geometric feature to highlight here is that unlike the shapes generated by the two other approaches, the shape of the deck's bottom chord computed by our CEM+FDM method is a deep continuous arch instead of a bulged ribbon. We hypothesize that the arched chord helps simplify the deck's load path and allows the cablenet to hold part of the deck loads more effectively to resolve the target constraints in a tensile internal stress state. The arch shape is found automatically by our approach.

4 Conclusion

In this paper, we explore how to couple the FDM and the CEM with automatic differentiation and gradient-based optimization to solve complex constrained form-finding problems. By sharing equilib-

rium and derivatives information between methods, we show that we can efficiently compute a constrained shape in static equilibrium for a cablenet-stayed bridge. In particular, we demonstrate that our approach converges to an optimal solution that matches target constraints with one order of magnitude higher accuracy than the shapes generated by either a single form-finding method or two methods used separately, and at no extra computation time. In the future, we want to formalize our approach into a general equilibrium framework that integrates more form-finding methods into the same computation pipeline and utilizes them to their strengths. We are also interested in measuring its robustness under different starting parameter values and in other constrained form-finding problems. Ultimately, we hope our work supports the practical design of lightweight structures by unifying the currently disjoint but rich palette of both traditional and modern form-finding methods available in the literature.

Acknowledgments

This work has been supported by the U.S. National Science Foundation under grant OAC-2118201 and the Institute for Data-Driven Dynamical Design (ID4).

References

[1] D. Veenendaal and P. Block, "An overview and comparison of structural form finding methods for general networks," *International Journal of Solids and Structures*, vol. 49, no. 26, pp. 3741–3753, Dec. 2012, ISSN: 0020-7683. DOI: 10.1016/j.ijsolstr.2012.08.008. (visited on 07/31/2020).

[2] 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, Jan. 1974, ISSN: 0045-7825. DOI: 10.1016/0045-7825(74)90045-0. (visited on 03/24/2020).

[3] P. Block and J. Ochsendorf, "Thrust network analysis: A new methodology for three-dimensional equilibrium," *Journal of the International Association for shell and spatial structures*, vol. 48, no. 3, pp. 167–173, 2007, ISSN: 1028-365X.

[4] P. O. Ohlbrock and P. D'Acunto, "A Computer-Aided Approach to Equilibrium Design Based on Graphic Statics and Combinatorial Variations," *Computer-Aided Design*, vol. 121, p. 102 802, Apr. 2020, ISSN: 0010-4485. DOI: 10.1016/J.CAD.2019.102802. (visited on 02/16/2020).

[5] F. Marmo and L. Rosati, "Reformulation and extension of the thrust network analysis," *Computers & Structures*, vol. 182, pp. 104–118, Apr. 2017, ISSN: 00457949. DOI: 10.1016/j.compstruc.2016.11.016. (visited on 10/03/2022).

[6] R. Maia Avelino, A. Iannuzzo, T. Van Mele, and P. Block, "Assessing the safety of vaulted masonry structures using thrust network analysis," *Computers & Structures*, vol. 257, p. 106 647, Dec. 2021, ISSN: 0045-7949. DOI: 10.1016/j.compstruc.2021.106647. (visited on 10/06/2021).

[7] R. Pastrana, P. O. Ohlbrock, T. Oberbichler, P. D'Acunto, and S. Parascho, "Constrained Form-Finding of Tension–Compression Structures using Automatic Differentiation," *Computer-Aided Design*, vol. 155, p. 103 435, Feb. 2023, ISSN: 0010-4485. DOI: 10.1016/j.cad.2022.103435. (visited on 11/09/2022).

[8] K. Saldana Ochoa, P. O. Ohlbrock, P. D'Acunto, and V. Moosavi, "Beyond typologies, beyond optimization: Exploring novel structural forms at the interface of human and machine intelligence," *International Journal of Architectural Computing*, p. 147 807 712 094 306, Jul. 2020, ISSN: 1478-0771, 2048-3988. DOI: 10.1177/1478077120943062. (visited on 08/08/2020).

[9] R. M. O. Pauletti and P. M. Pimenta, "The natural force density method for the shape finding of taut structures," *Computer Methods in Applied Mechanics and Engineering*, vol. 197, no. 49, pp. 4419–4428, Sep. 2008, ISSN: 0045-7825. DOI: 10.1016/j.cma.2008.05.017. (visited on 07/15/2021).

[10] M. Miki and K. Kawaguchi, "Extended Force Density Method for Form-Finding of Tension Structures," *Journal of the International Association for Shell and Spatial Structures*, vol. 51, no. 4, p. 13, 2010.

[11] R. Pastrana, D. Oktay, R. P. Adams, and S. Adriaenssens, "JAX FDM: A differentiable solver for inverse form-finding," in *ICML 2023 Workshop on Differentiable Almost Everything: Differentiable Relaxations, Algorithms, Operators, and Simulators*, 2023. [Online]. Available: https://openreview.net/forum?id=Uu9OPgh24d.

[12] A. G. Baydin, B. A. Pearlmutter, A. A. Radul, and J. M. Siskind, "Automatic differentiation in machine learning: A survey," *arXiv:1502.05767 [cs, stat]*, Feb. 2018. arXiv: 1502.05767 [cs, stat]. [Online]. Available: http://arxiv.org/abs/1502.05767 (visited on 11/26/2020).

[13] J. Nocedal, "Updating quasi-Newton matrices with limited storage," *Mathematics of Computation*, vol. 35, no. 151, pp. 773–773, Sep. 1980, ISSN: 0025-5718. DOI: 10.1090/S0025-5718-1980-0572855-7. (visited on 03/17/2021).

[14] J. Bradbury *et al.*, *JAX: Composable transformations of Python+NumPy programs*, 2018. [Online]. Available: http://github.com/google/jax.

[15] M. Blondel *et al.*, *Efficient and Modular Implicit Differentiation*, Oct. 2022. DOI: 10.48550/arXiv.2105.15183. arXiv: 2105.15183 [cs, math, stat]. (visited on 10/03/2023).