

FDMopt: force density method for optimal geometry and topology of trusses

Kazuki Hayashi^a, Makoto Ohsaki^a

^a*Department of Architecture and Architectural Engineering, Graduate School of Engineering, Kyoto University, Kyoto, Japan*

Abstract

This paper presents a new efficient tool for simultaneous optimization of topology and geometry of truss structures. Force density method is applied to formulate optimization problem to minimize compliance under constraint on total structural volume, and objective and constraint functions are expressed as explicit functions of force density only. This method does not need constraints on nodal locations to avoid coalescent nodes, and enables to generate optimal solutions with a variety in topology and geometry. Furthermore, for the purpose of controlling optimal shapes, tensor product Bézier surface is introduced as a design surface. The optimization problem is solved using sensitivity coefficients and the optimizer is compiled as a component compatible with Grasshopper, an algorithmic modeling plug-in for Rhinoceros, which is a popular 3D modeling software. Efficiency and accuracy of the proposed method are demonstrated through two numerical examples of semi-cylindrical and semi-spherical models.

Keywords: Truss, Latticed shell, Force density method, Simultaneous optimization of topology and geometry, Tensor product Bézier surface, Grasshopper, Interactive design

1. Introduction

There are three major categories in the field of truss optimization. First one is size optimization, where cross-sectional areas of members are considered to be design variables. From 1960s, size optimization is extensively studied for optimum design under stress constraints, which is called *fully stressed design* [1], where the stress of any member is equal to its upper or lower bound for at least one of the specified loading conditions.

Although size optimization has become popular in practical applications, it is difficult to find unexpected solutions, since the truss configuration is fixed and only cross-sectional areas are optimized.

Second category is topology optimization, which controls the connectivity of members. Topology optimization is also a well established field of research and theoretical works are summarized in, e.g., Refs. [2] and [3]. Above all, *ground structure method* (GSM) is widely used for topology optimization; it starts from a highly connected structure called *ground structure* and eliminate unnecessary members [4].

The last category is geometry optimization, which is also called shape optimization, and controls nodal locations to change overall truss geometry. Although

numerous mathematical programming approaches have been utilized for optimizing geometry [5, 6], they inevitably need constraints on nodal locations to prevent numerical difficulty due to existence of extremely short members, called *melting nodes* [7] or *coalescent nodes* [8]. Therefore, there is little possibility to obtain a sparse optimal topology by simply setting nodal coordinates as design variables. We omit the description of size optimization, since both processes of topology and geometry optimization usually includes size optimization [9].

To obtain optimal topology and geometry of trusses simultaneously, *growth method* is one of the well studied approaches [10, 11]. It starts from a relatively sparse set of nodes and members, and add them by heuristics. Although growth method yields an optimal solution of sparse topology and geometry, the addition of nodes and members does not satisfy any theoretically defined optimality criterion. By contrast, it needs substantial computational cost if the GSM is applied to obtain an optimal geometry with acceptable precision, because nodal locations are fixed in this method; and accordingly, a large number of nodes and members are needed to optimize nodal locations.

Another approach is a hybrid method of the three

51 types of optimization methods above. The frequently 103
52 applied method is an alternating optimization of topol- 104
53 ogy and geometry; however such two-level algorithms 105
54 need too much computational cost to optimize large 106
55 scale trusses [9]. In addition, quality of the result is 107
56 hard to evaluate, since alternating optimization easily 108
57 converges to a non-optimal solution. 109

58 Topology optimization and geometry optimization 110
59 can be simultaneously conducted by setting cross- 111
60 sectional areas of members and nodal coordinates as de- 112
61 sign variables in a single optimization problem. This is 113
62 called *simultaneous optimization* of topology and geom- 114
63 etry, which is very difficult to solve, because it is nec- 115
64 essary to modify the topology by removing coalescent 116
65 nodes while varying the nodal locations. Achtziger [8] 117
66 presented a simultaneous optimization method based on 118
67 implicit programming. Although it always converges to 119
68 an optimal solution because a mathematical program- 120
69 ming approach is used, side constraints are needed to 121
70 avoid melting nodes. 122

71 Latticed shells are one of the large scale structures 123
72 composed of a number of truss members. In determin- 124
73 ing shapes and topologies of latticed shells, not only the 125
74 designer’s preference but also mechanical properties of 126
75 the members play an important role. Thus, its design 127
76 problem is formulated as a multiobjective optimization 128
77 problem considering geometrical and mechanical prop- 129
78 erties, and parametric surfaces are often used to design 130
79 latticed shells with non-standard shapes [9]. Ramm *et* 131
80 *al.* [12] used Bézier surfaces for geometry optimization 132
81 of shells. Ohsaki *et al.* [13] formulated an optimiza- 133
82 tion problem for a double-layer space truss to minimize 134
83 compliance under constraints on total structural volume, 135
84 where triangular Bézier patch is used to define the ge- 136
85 ometry of the upper layer surface. 137

86 Owing to a large number of researches in structural 138
87 optimization, there is an increasing number of practical 139
88 optimization tools available to designers and engineers. 140
89 Especially, structural optimization in Grasshopper [14] 141
90 is widely used among architects and structural engi- 142
91 neers. Grasshopper is an algorithmic modeling plug-in 143
92 for Rhinoceros, which is a 3D modeling software [15]. 144
93 As a general approach to optimizing structures within 145
94 the framework of Grasshopper, the users usually com- 146
95 bine components of structural analysis and optimiza- 147
96 tion independently. Therefore, evolutionary optimiza- 148
97 tion tools such as Galapagos [16] and Goat [17] are fre- 149
98 quently used, because they are applicable to most of the 150
99 optimization problems. 151

100 According to Bradner *et al.* [18], an optimal solution 148
101 obtained by an optimization tool is often used as the 149
102 starting point for design exploration, not the end prod- 150

uct. Thus, it is important that the optimizer generates 103
diverse optimal solutions, and simultaneous optimiza- 104
tion of topology and geometry has potential to present 105
diverse candidate designs. 106

107 However, because of the difficulties mentioned 108
above, there is no practical tool to simultaneously op- 109
timize topology and geometry of trusses. Even the 110
optimization process is somehow constructed within 111
Grasshopper by combining a structural analyzer and an 112
evolutionary optimizer, there is little possibility to ob- 113
tain feasible solutions because the complexity of the 114
problem is difficult to resolve in a simple manner, and 115
the solutions do not always satisfy any optimality crite- 116
rion. Therefore, it is necessary to develop a “package” 117
of the framework to conduct the simultaneous optimiza- 118
tion. 119

120 This paper aims to develop a Grasshopper component 121
for simultaneous optimization of topology and geome- 122
try of trusses based on the force density method (FDM) 123
proposed by Ohsaki and Hayashi [19]. The optimization 124
problem is formulated as functions of force densities 125
only; therefore, computational cost can be drastically 126
reduced compared with previous methods where nodal 127
coordinates and cross-sectional areas are assigned as de- 128
sign variables. Moreover, numerical difficulties due to 129
melting nodes are successfully avoided by simply set- 130
ting upper bounds for design variables, and thus various 131
optimal solutions of topology and geometry are gener- 132
ated from a relatively sparse initial ground structure. We 133
further introduce a free-form design surface to control 134
nodal locations. Once a design surface is specified by an 135
architectural designer, the optimizer moves nodes along 136
and on the surface. 137

138 The paper is organized as follows. In section 2, we 139
summarize formulation of the optimization problem. In 140
section 3, compilation of the Grasshopper component 141
which “packages” the optimization problem is outlined. 142
The overall workflow to optimize the model is also de- 143
scribed in this section. In section 4, two numerical ex- 144
amples are demonstrated to evaluate the optimizer’s so- 145
lutions in terms of their feasibility. In section 5, we 146
close the paper with concluding remarks. 147

145 2. Optimization problem

148 We focus on a minimization problem of compliance 149
under constraint on total structural volume and on nodal 150
locations such that they are on a prescribed tensor prod-
uct Bézier surface. Note that all the following variables
are described as functions of force density only.

2.1. Force density method

Outline of the FDM for simultaneous optimization is presented here for completeness of the paper. See Ohsaki and Hayashi [19] for details. Free nodal coordinates are formulated as functions of force density. Force density of member i is defined with respect to the axial force N_i and the length L_i as

$$q_i = \frac{N_i}{L_i} \quad (1)$$

Consider a truss with m members and n nodes. If member i connects nodes j and k , then components of connectivity matrix $\mathbf{C} \in \mathbb{R}^{m \times n}$ are defined as

$$C_{ij} = -1, C_{ik} = 1 \quad (i = 1, \dots, m) \quad (2)$$

and the remaining components are 0. Using \mathbf{C} and the force density vector $\mathbf{q} \in \mathbb{R}^m$, the force density matrix $\mathbf{Q} \in \mathbb{R}^{n \times n}$ is defined as

$$\mathbf{Q} = \mathbf{C}^T \text{diag}(\mathbf{q})\mathbf{C} \quad (3)$$

The same matrix \mathbf{Q} can be obtained for components of force densities in x -, y -, and z - directions, because the ratios of axial force to member length are all the same. \mathbf{Q} is re-assembled into $\tilde{\mathbf{Q}} \in \mathbb{R}^{3m \times 3n}$ so that the components of free coordinates precede those of fixed coordinates as

$$\tilde{\mathbf{Q}} = \begin{pmatrix} \mathbf{Q}_{\text{free}}^x & \mathbf{0} & \mathbf{0} & \mathbf{Q}_{\text{link}}^x & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{Q}_{\text{free}}^y & \mathbf{0} & \mathbf{0} & \mathbf{Q}_{\text{link}}^y & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{Q}_{\text{free}}^z & \mathbf{0} & \mathbf{0} & \mathbf{Q}_{\text{link}}^z \\ \mathbf{Q}_{\text{link}}^{xT} & \mathbf{0} & \mathbf{0} & \mathbf{Q}_{\text{fix}}^x & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{Q}_{\text{link}}^{yT} & \mathbf{0} & \mathbf{0} & \mathbf{Q}_{\text{fix}}^y & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{Q}_{\text{link}}^{zT} & \mathbf{0} & \mathbf{0} & \mathbf{Q}_{\text{fix}}^z \end{pmatrix} \quad (4)$$

Note that we use the term *fixed node* to express nodes that do not move in the optimization process. Therefore, fixed nodes include not only pin-supported but also loaded ones. Let n_{free} and n_{fix} denote the numbers of free and fixed DOFs satisfying

$$n_{\text{free}} + n_{\text{fix}} = 3n \quad (5)$$

Then, matrices $(\mathbf{Q}_{\text{free}}^x, \mathbf{Q}_{\text{free}}^y, \mathbf{Q}_{\text{free}}^z)$, $(\mathbf{Q}_{\text{link}}^x, \mathbf{Q}_{\text{link}}^y, \mathbf{Q}_{\text{link}}^z)$, and $(\mathbf{Q}_{\text{fix}}^x, \mathbf{Q}_{\text{fix}}^y, \mathbf{Q}_{\text{fix}}^z)$ are combined to $\tilde{\mathbf{Q}}_{\text{free}} \in \mathbb{R}^{n_{\text{free}} \times n_{\text{free}}}$, $\tilde{\mathbf{Q}}_{\text{link}} \in \mathbb{R}^{n_{\text{free}} \times n_{\text{fix}}}$, and $\tilde{\mathbf{Q}}_{\text{fix}} \in \mathbb{R}^{n_{\text{fix}} \times n_{\text{fix}}}$, respectively.

If the force densities of all members and fixed nodal coordinates $\mathbf{X}_{\text{fix}} \in \mathbb{R}^{n_{\text{fix}}}$ are specified, then the free nodal coordinates $\mathbf{X}_{\text{free}} \in \mathbb{R}^{n_{\text{free}}}$ are obtained from the following system of linear equations:

$$\tilde{\mathbf{Q}}_{\text{free}}\mathbf{X}_{\text{free}} = -\tilde{\mathbf{Q}}_{\text{link}}\mathbf{X}_{\text{fix}} \quad (6)$$

Therefore, \mathbf{X}_{free} is a function of \mathbf{q} .

2.2. Objective and constraint functions

Consider a problem for minimizing compliance under total structural volume. The solution to the optimization problem is a statically determinate truss with the same absolute value of axial stress $\bar{\sigma}$ for all members [8, 20]. Hence, the cross-sectional area of member i is expressed as

$$A_i = \frac{|q_i|L_i}{\bar{\sigma}} \quad (7)$$

If member i connects nodes j and k , the square of L_i is given as

$$L_i^2 = (\mathbf{X}_k - \mathbf{X}_j)^T(\mathbf{X}_k - \mathbf{X}_j) \quad (8)$$

where $\mathbf{X}_j \in \mathbb{R}^3$ and $\mathbf{X}_k \in \mathbb{R}^3$ are the position vectors of nodes j and k , respectively.

Let Young's modulus E and a very small positive number c be given. Then the compliance can be expressed as

$$F = \sum_{i=1}^m \frac{\bar{\sigma}L_i^2 \sqrt{q_i^2 + c}}{E} \quad (9)$$

In (9), the absolute value of force density $|q_i|$ is substituted by $\sqrt{q_i^2 + c}$ for the purpose of smoothness of the objective function.

Fig. 1 illustrates the variation of $f_i(q_i) = \bar{\sigma}L_i^2|q_i|/E$ with and without smoothing. If $|q_i|$ is used for the objective function without smoothing, the sensitivity coefficient is discontinuous at $q_i = 0$, which causes difficulty of convergence. On the other hand, the sensitivity coefficient becomes continuous around 0 by introducing smoothing parameter c , which should be small enough to neglect its effect on the compliance value.

Since the product of the total structural volume and the compliance is independent of $\bar{\sigma}$, the total structural volume can be calculated after minimizing the compliance with arbitrary positive value of $\bar{\sigma}$ [19].

Define $\mathbf{R} \in \mathbb{R}^{n_{\text{fix}}}$ as the vector of reaction forces corresponding to \mathbf{X}_{fix} , which is obtained from

$$\mathbf{R} = \tilde{\mathbf{Q}}_{\text{link}}^T \mathbf{X}_{\text{free}} + \tilde{\mathbf{Q}}_{\text{fix}} \mathbf{X}_{\text{fix}} \quad (10)$$

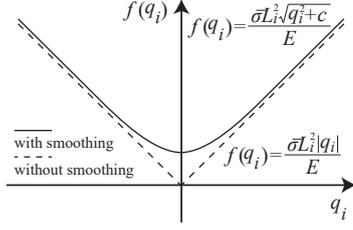


Figure 1: Sensitivity coefficient with and without smoothing.

209 In the optimization problem, the loading condition must
210 be considered by prescribing reaction forces at loaded
211 nodes as

$$\sum_{i \in \mathbf{I}_R} (R_i - \bar{P}_i)^2 = 0 \quad (11)$$

212 where \mathbf{I}_R is the set of indices of reaction forces to be
213 specified and \bar{P}_i is the specified load value.

214 We further add constraint on nodal locations using a
215 design surface. If z -coordinate of a point on the surface
216 can be expressed as an explicit function of x - and y -
217 coordinates, the constraint can be expressed as

$$z_i = f(x_i, y_i) \quad (i = 1, \dots, n) \quad (12)$$

218 Thus, the optimization problem can be formulated as

$$\text{minimize } F(\mathbf{q}) = \sum_{i=1}^m \frac{\bar{\sigma} L_i^2 \sqrt{q_i^2 + c}}{E} \quad (13a)$$

$$\text{subject to } \sum_{i \in \mathbf{I}_R} (R_i - \bar{P}_i)^2 = 0 \quad (13b)$$

$$z_i = f(x_i, y_i) \quad (i = 1, \dots, n) \quad (13c)$$

$$0.0 \leq x_i \leq 1.0 \quad (i = 1, \dots, n) \quad (13d)$$

$$0.0 \leq y_i \leq 1.0 \quad (i = 1, \dots, n) \quad (13e)$$

$$q_k^L \leq q_k \leq q_k^U \quad (k = 1, \dots, m) \quad (13f)$$

219 where q_k^L and q_k^U are the lower and upper bounds for q_k ,
220 respectively. Necessity of constraints (13d) and (13e)
221 will be explained in the next sub-section.

2.3. Explicit expression of Bézier surface

223 According to constraint (13c), the design surface
224 need to be expressed as an explicit function. A tensor
225 product Bézier surface of order $M \times N$ can be expressed
226 with parameters $u, v \in [0.0, 1.0]$ as

$$\mathbf{S}(u, v) = \sum_{i=0}^M \sum_{j=0}^N B_i^M(u) B_j^N(v) \mathbf{P}_{ij} \quad (14)$$

227 where $\mathbf{P}_{ij} \in \mathbb{R}^3$ is the position vector of a control point
228 to define the design surface. The functions $B_i^M(u)$ and
229 $B_j^N(v)$ are the Bernstein basis polynomials in u - and v -
230 directions, respectively, which are written as

$$B_i^M(u) = \binom{M}{i} u^i (1-u)^{M-i} \quad (15a)$$

$$B_j^N(v) = \binom{N}{j} v^j (1-v)^{N-j} \quad (15b)$$

If \mathbf{P}_{ij} is described as

$$\mathbf{P}_{ij} = (i/M, j/N, b_{ij})^T \quad (i = 0, \dots, M, j = 0, \dots, N, b_{ij} \in \mathbb{R}) \quad (16)$$

232 then z -coordinate of any arbitrary point on the tensor
233 product Bézier surface can be expressed as an explicit
234 function of its x - and y -coordinates, since the following
235 equation is satisfied [21]:

$$\mathbf{S}(u, v) = \left(u, v, \sum_{i=0}^M \sum_{j=0}^N B_i^M(u) B_j^N(v) b_{ij} \right)^T \quad (17)$$

236 If the surface is scaled to satisfy $x = u$ and $y = v$, then
237 (17) can be re-written as

$$z = f(x, y) = \sum_{i=0}^M \sum_{j=0}^N B_i^M(x) B_j^N(y) b_{ij} \quad (18)$$

238 From (18), every control point \mathbf{P}_{ij} must be placed at
239 equal intervals in x - and y -directions and its x - and y -
240 coordinates must be within the range of $[0.0, 1.0]$. The
241 latter condition requires constraints (13d) and (13e) in
242 the optimization problem.

2.4. Sensitivity analysis

243 To reduce computational cost for solving (13), sensi-
244 tivity coefficients of objective and constraint functions
245 are analytically obtained in this sub-section. The ob-
246 jective function (13a) is differentiated with respect to q_l
247 as
248

$$\frac{\partial F(\mathbf{q})}{\partial q_l} = \frac{\bar{\sigma} q_l L_l^2}{E \sqrt{q_l^2 + c}} + \sum_{i=1}^m \left(\frac{\bar{\sigma} \sqrt{q_i^2 + c}}{E} \cdot \frac{\partial L_i^2}{\partial q_l} \right) \quad (19)$$

249 From (8), the sensitivity coefficient of L_i^2 with respect
250 to q_l is obtained as

$$\frac{\partial L_i^2}{\partial q_l} = 2(\mathbf{X}_k - \mathbf{X}_j)^T \cdot \frac{\partial(\mathbf{X}_k - \mathbf{X}_j)}{\partial q_l} \quad (20)$$

251 Differentiation of Eq. (6) with respect to q_l leads to

$$\tilde{\mathbf{Q}}_{\text{free}} \frac{\partial \mathbf{X}_{\text{free}}}{\partial q_l} + \frac{\partial \tilde{\mathbf{Q}}_{\text{free}}}{\partial q_l} \mathbf{X}_{\text{free}} = - \frac{\partial \tilde{\mathbf{Q}}_{\text{link}}}{\partial q_l} \mathbf{X}_{\text{fix}} \quad (21)$$

252 Therefore, sensitivity coefficient of objective function
253 with respect to design variable can be analytically ob-
254 tained from Eq. (19) to (21).

255 Next, Eq. (13b) is differentiated as

$$\sum_{i \in \mathbf{I}_R} \left(2(R_i - \bar{P}_i) \frac{\partial R_i}{\partial q_l} \right) = 0 \quad (22)$$

256 Sensitivity coefficients of reaction forces are computed
257 from (10) as

$$\frac{\partial \mathbf{R}}{\partial q_l} = \frac{\partial \tilde{\mathbf{Q}}_{\text{link}}^T}{\partial q_l} \mathbf{X}_{\text{free}} + \tilde{\mathbf{Q}}_{\text{link}}^T \frac{\partial \mathbf{X}_{\text{free}}}{\partial q_l} + \frac{\partial \tilde{\mathbf{Q}}_{\text{fix}}}{\partial q_l} \mathbf{X}_{\text{fix}} \quad (23)$$

258 Sensitivity coefficient of $f(x_k, y_k)$ is derived by differen-
259 tiating Eq. (18) at $(x, y) = (x_k, y_k)$ as

$$\begin{aligned} & \frac{\partial f(x_k, y_k)}{\partial q_l} \\ &= \sum_{i=0}^M \sum_{j=0}^N \left(\frac{\partial B_i^M(x_k)}{\partial q_l} B_j^N(y_k) + B_i^M(x_k) \frac{\partial B_j^N(y_k)}{\partial q_l} \right) b_{ij} \end{aligned} \quad (24)$$

260 From (15a) and (15b), differentiation of Bernstein ba-
261 sis polynomial with respect to q_l with $u = x_k$ and $v = y_k$
262 leads to

$$\begin{aligned} & \frac{\partial B_i^M(x_k)}{\partial q_l} \\ &= \binom{M}{i} \left(i x_k^{i-1} (1-x_k)^{M-i} - (M-i) x_k^i (1-x_k)^{M-i-1} \right) \frac{\partial x_k}{\partial q_l} \end{aligned} \quad (25a)$$

$$\begin{aligned} & \frac{\partial B_j^N(y_k)}{\partial q_l} \\ &= \binom{N}{j} \left(j y_k^{j-1} (1-y_k)^{N-j} - (N-j) y_k^j (1-y_k)^{N-j-1} \right) \frac{\partial y_k}{\partial q_l} \end{aligned} \quad (25b)$$

263 Sensitivity coefficients of nodal coordinates in Eqs.
264 (20), (23), and (25) are derived by solving the system of
265 linear equations (21) for $\partial \mathbf{X}_{\text{free}} / \partial q_l$.

266 3. System architecture

267 In this section, the process of developing the opti-
268 mizer in Grasshopper is described. In optimizing
269 trusses, initial ground structure, support and loading
270 conditions, and additional constraints must be translated
271 to numerical data to be incorporated to the optimization
272 program. It takes much time and is prone to mistakes
273 if we conduct the translation manually. Instead, we de-
274 veloped a Grasshopper component to automatically ex-
275 tract information which is necessary for solving the op-
276 timization problem.

277 Fig. 2 shows the Grasshopper component coded in
278 C# with the aid of Grasshopper SDK. This component
279 is capable of handling geometry classes of Rhinoceros
280 directly. However, we dare to set *supporting condition*
281 *Sp* as a text input instead of *Point* class, to facilitate re-
282 leasing the boundary condition; for example, the user
283 only need to write “3, xy” when releasing the third node
284 in z -direction. If the component succeeds to collect the
285 data from the required fields, it triggers an initialization
286 *method*, a code block containing a series of statements.
287 The component converts software-specific geometry in-
288 formation into numerical data through the framework of
289 RhinoCommon API so that the optimization algorithm
290 is able to handle it. The user is also able to check the
291 numerical data because it is saved in text format in the
292 local storage. Note that the initialization *method* does
293 not include optimization procedure, because the opti-
294 mization should start after finishing all the input setting.
295 We further added six optional inputs. L, x, y, z are pos-
296 itive real numbers to control shape change from the ini-
297 tial solution. Though *design surface S* is also an op-
298 tional input for shape controlling, it can be specified by
299 *Surface* class, which is one of basic geometry types of
300 Rhinoceros. Once the component collects the *Surface*
301 geometry, it retrieves locations of its controlling points
302 in order to calculate Eqs. (18) and (24) in the optimiza-
303 tion process. I is a random seed to randomize initial
304 force densities; by altering this value, the user is able to
305 obtain different solutions from the same initial ground
306 structure.

307 Once the component is double-clicked, it starts call-
308 ing an optimization program compiled in FORTRAN,
309 where SNOPT ver. 7.2, an SQP solver is incorporated
310 [22]. The FORTRAN program randomizes the initial
311 force densities based on the prescribed random seed,
312 conducts the optimization, and returns an optimal so-
313 lution back to the Grasshopper component. Owing to
314 geometry processing libraries and a graphical interface
315 of Rhinoceros, the optimal solutions can be easily visu-
316 alized, which offers users real-time feedback of the op-

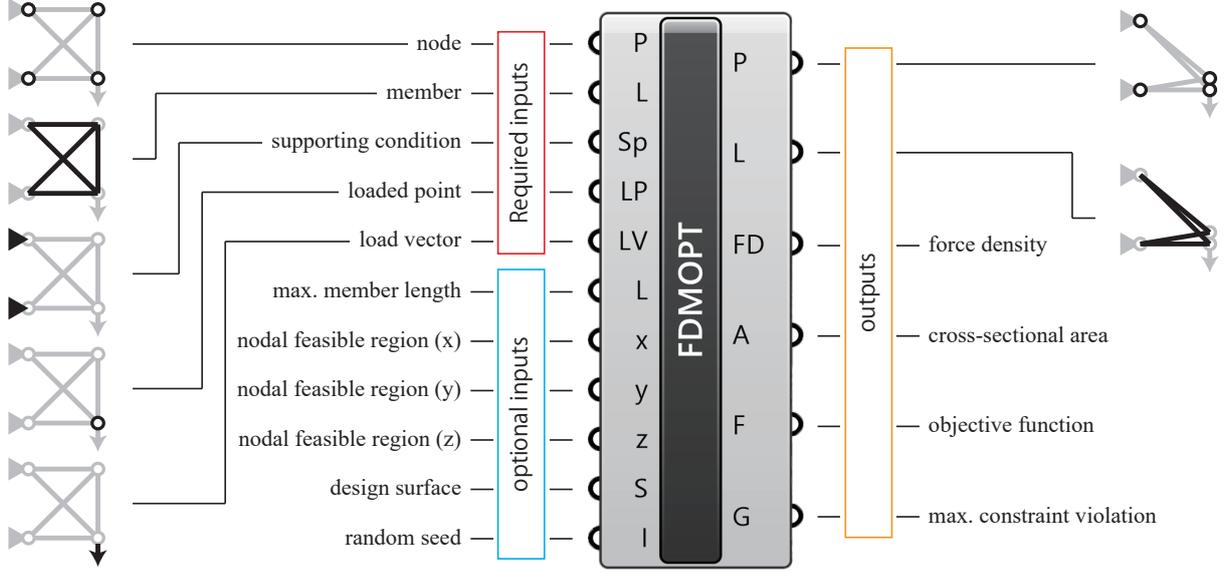


Figure 2: Compiled Grasshopper component.

317 timal geometry and topology of trusses. The whole optimization workflow is illustrated in Fig. 3. This framework allows the user to conduct optimization with the proposed method, review the result, and change the conditions of the initial model for design exploration easily and interactively.

323 4. Numerical examples

324 In this section, we demonstrate our proposed method through numerical examples. In the optimization process, Young's modulus E , the absolute value of axial stress $\bar{\sigma}$, and the smoothing approximation factor c are first set to be 1.0 N/mm^2 , 1.0 N/mm^2 , and $1.0 \times 10^{-6} \text{ N}^2/\text{m}^2$, respectively. Note that the cross-sectional areas and the objective function value are re-scaled such that $E = 2.0 \times 10^5 \text{ [N/mm}^2\text{]}$, $\bar{\sigma} = 200 \text{ [N/mm}^2\text{]}$, and the maximum volume $\bar{V} = 0.01 \text{ [m}^3\text{]}$ after obtaining the optimal solutions.

Let q_{init}^i ($i = 1, \dots, m$) and \bar{r} (> 0) denote initial force density of the i th member for the initial model and the random seed. Then every force density is randomized within the following range before starting optimization:

$$q_i^{\text{start}} \in [q_i^{\text{init}} - \bar{r}, q_i^{\text{init}} + \bar{r}] \quad (26)$$

Empirically, the order of \bar{r} should be less than that of initial force densities to stabilize the optimization. Considering the maximum absolute value of initial force density is around 10.0 N/m in the following examples, \bar{r} is

set to be 1.0 N/m . After randomizing the initial force densities, the lower and upper bounds are defined as

$$q_i^L = q_i^{\text{start}} - \bar{d} \quad (27a)$$

$$q_i^U = q_i^{\text{start}} + \bar{d} \quad (27b)$$

334 A small value is preferable for \bar{d} in view of convergence; however relatively large value compared with \bar{r} and initial force densities must be assigned in order to ensure wide variable range. We set $\bar{d} = 1.0 \times 10^2 \text{ [N/m]}$ in the following examples.

335 Optimization is conducted 100 times for the 100 different initial sets of initial force densities to select the best solution with the least objective function value. If the maximum violation of constraints is more than 1.0×10^{-4} , the solution is rejected as infeasible. We use a PC with Intel Core i9-7900X [3.30 GHz/10 Core] processor in the following examples.

346 4.1. Semi-cylindrical latticed shell

347 The first example is a semi-cylindrical double-layer latticed shell with 46 nodes and 225 members, as shown in Fig. 4. The bottom four corner nodes are pin-supported and all the upper nodes are subjected to downward unit loads 1.0 N . Note again that these 25 supported or loaded nodes cannot move in the optimization process. A semi-cylindrical quadratic Bézier surface is introduced as a design surface which interpolates the 21 free nodes.

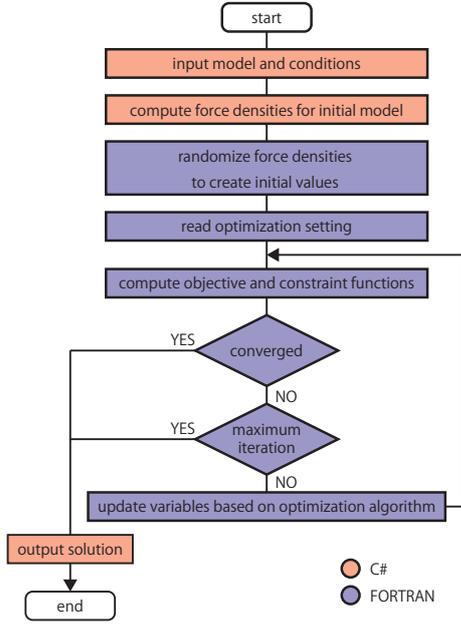


Figure 3: Overall flowchart conducted in the optimizer.

356 After 100 trials, 73 valid solutions are obtained. The
 357 maximum, median, minimum, average values and stan-
 358 dard deviation of F for the 73 valid solutions are listed
 359 in Table 1. It takes 115 seconds for each trial on av-
 360 erage. The optimal solution with the least value of
 361 $F = 73.104[\text{N}\cdot\text{m}]$ is shown in Fig. 5, and their nodal
 362 locations are listed in Table. 2. The pair of nodes 22
 363 and 23 is coalescent within the range of 0.01 m to ge-
 364 nerate the simpler shape with 45 nodes.

365 We also conducted optimization without smoothing
 366 of objective function, and obtained only 21 feasible so-

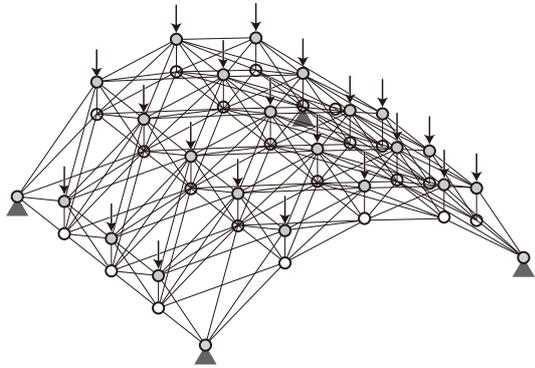
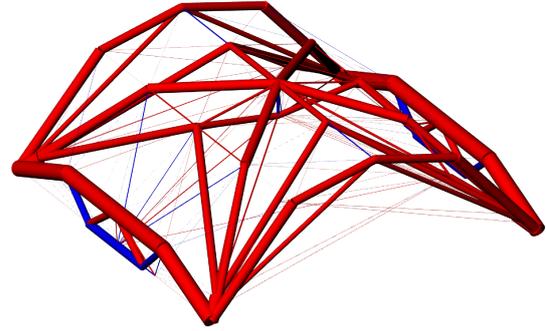
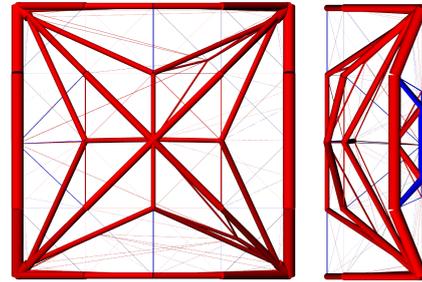


Figure 4: Initial ground structure of a semi-cylindrical lattices shell.



(a) Isometric view.



(b) Plan and elevations.

Figure 5: The optimal solution of semi-cylindrical latticed shell ($F = 73.104[\text{N}\cdot\text{m}]$).

367 lutions out of 100 initial solutions. Therefore, smoothing
 368 of objective function is crucial to improve optimiza-
 369 tion performance.

Table 1: Statistical results of F [N·m] for 73 converged solutions of semi-cylindrical latticed shell.

| Max. | Median | Min. | Average | Std. dev. |
|---------|--------|--------|---------|-----------|
| 112.109 | 75.084 | 73.104 | 76.548 | 5.885 |

370 Since the optimal solution to Problem (13) includes
 371 overlapped nodes and very thin members, and the con-
 372 figuration is obscure, we improve the optimization re-
 373 sult with the following steps. First, overlapped nodes
 374 and members are unified in the optimal solution to Prob-
 375 lem (13). Let m^* denote the number of members after
 376 unifying overlapped nodes and members, then the cross-
 377 sectional areas of members are optimized by solving the
 378 following problem:

$$\text{minimize } F(\mathbf{A}) = \sum_{i=1}^{m^*} \frac{N_i^2 L_i}{EA_i} \quad (28a)$$

$$\text{subject to } \sum_{i=1}^{m^*} A_i L_i \leq \bar{V} \quad (28b)$$

$$A_i^L \leq A_i \leq A_i^U \quad (i = 1, \dots, m^*) \quad (28c)$$

379 The lower bound A_i^L for A_i is 1.0×10^{-3} [m²], and the
 380 member with $A_i \leq 1.0 \times 10^{-2}$ [m²] is eliminated after op-
 381 timization. Finite difference method is applied for com-
 382 puting sensitivity coefficients, because Problem (28) is
 383 solved only once.

384 The improved solution is shown in Fig. 6. The
 385 compliance F slightly increased to 73.259 N·m. This
 386 implies that the influence of re-optimization is trivial
 387 and FDMopt produced a sufficiently converged solution.
 388 The number of members is reduced to 85. Although
 389 there is no constraint on symmetry, almost symmetric
 390 optimal shape is obtained.

391 4.2. Semi-spherical latticed shell

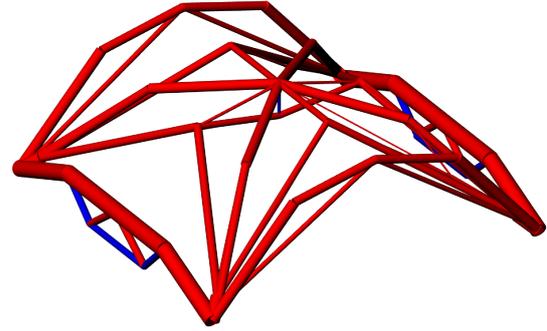
392 The second example is a semi-spherical double-layer
 393 latticed shell as shown in Fig. 7 with the same number
 394 of nodes and members as the first example. The bottom
 395 four corner nodes are pin-supported and all the upper
 396 nodes are subjected to downward unit loads. As well as
 397 the first example, a semi-spherical quadratic Bézier sur-
 398 face is introduced as a design surface which interpolates
 399 the 21 free nodes.

400 After 100 trials, 84 valid solutions are obtained. The
 401 maximum, median, minimum, average values and stan-
 402 dard deviation of F for the 84 valid solutions are listed
 403 in Table 3. It takes 481 seconds for each trial on aver-
 404 age. The optimal solution with the least value of
 405 $F = 37.646$ [N·m] is shown in Fig. 8, and their nodal
 406 locations are listed in Table. 4. The distance between
 407 nodes 7 and 12 is 0.007 m which is very small.

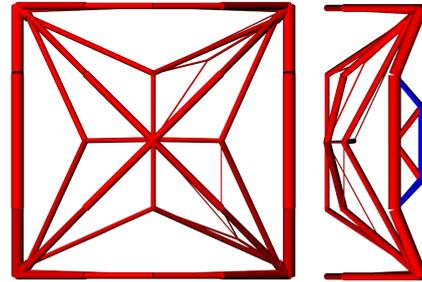
408 We further solve Problem (28) in the same manner as
 409 the first example, and the result is shown in Fig. 9. The
 410 compliance F becomes 37.824 [N·m], and the number
 411 of members is reduced from 225 to 83.

412 5. Conclusion

413 We proposed an interactive and integrated design ap-
 414 proach to truss design by developing a Grasshopper
 415 component for the simultaneous optimization of geom-
 416 etry and topology of trusses. The numerical difficulty
 417 due to melting nodes can be successfully avoided us-
 418 ing force density as design variable, which contributes



(a) Isometric view.



(b) Plan and elevations.

Figure 6: The solution after re-optimization of semi-cylindrical latticed shell ($F = 73.259$ [N·m]).

419 to generating optimal solutions with a variety of geom-
 420 etry and topology from a relatively sparse initial ground
 421 structure. Moreover, a tensor product Bézier surface is
 422 successfully incorporated as a design surface of the opti-
 423 mizer, which enables reflection of the user's shape pref-
 424 erence.

425 We further introduced sensitivity analysis in the solu-
 426 tion process of the optimization problem to reduce the
 427 computational cost. Owing to the sensitivity analysis
 428 and the less number of design variables compared with
 429 previous researches, the optimizer is able to yield a so-
 430 lution in a short time even for relatively complex models
 431 demonstrated in the examples.

432 Although the design surface is restricted to a tensor
 433 product Bézier surface whose control points are placed
 434 at equal intervals in x - and y -directions, this restriction
 435 is expected to be alleviated by using geometric transfor-
 436 mation in future research.

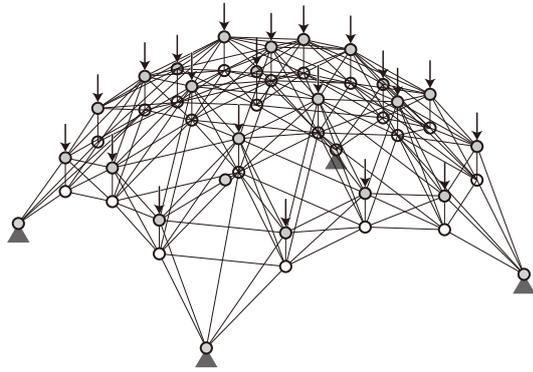
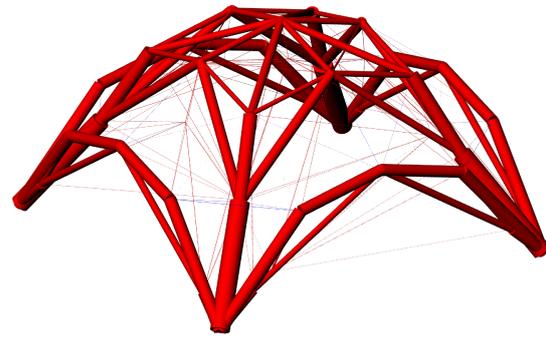
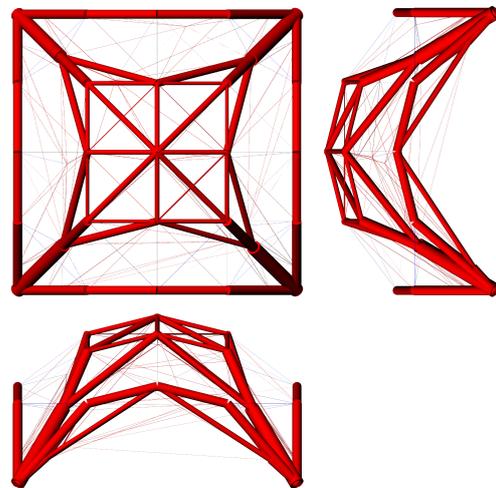


Figure 7: Initial ground structure of a semi-spherical latticed shell.



(a) Isometric view.



(b) Plan and elevations.

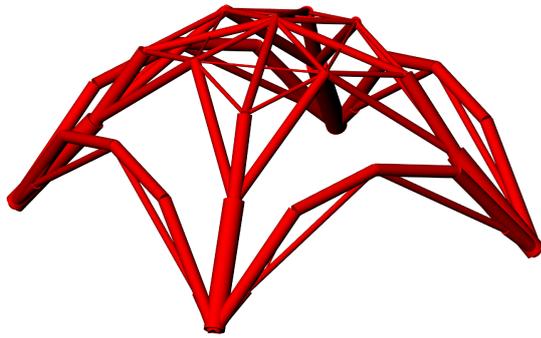
Figure 8: The optimal solution of semi-spherical latticed shell ($F = 37.646[\text{N}\cdot\text{m}]$).

Acknowledgement

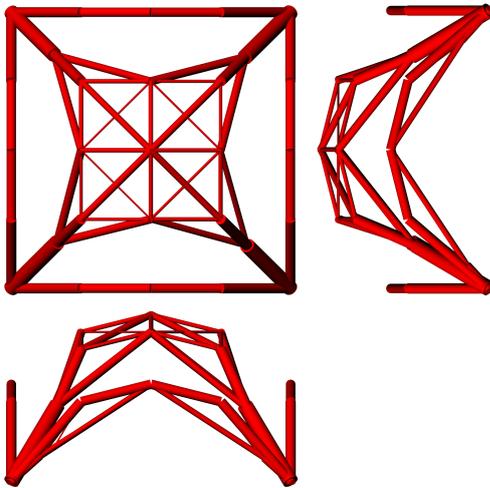
This work is supported by the “SPIRITS” program of Kyoto University, Grant-in-Aid for JSPS Research Fellow Grant Number 18J21456 and JSPS KAKENHI Grant Number 16H03014. The authors wish to express their appreciation to Professor Caitlin Mueller at Massachusetts Institute of Technology for her valuable comments.

References

- [1] R. Razani, Behavior of fully stressed design of structures and its relationship to minimum-weight design, *AIAA Journal* 3 (1965) 2262–2268.
- [2] U. Kirsch, Optimal topologies of truss structures, *Computer Methods in Applied Mechanics and Engineering* 72 (1989) 15–28.
- [3] M. P. Bendsøe, O. Sigmund, *Topology Optimization: Theory, Methods and Applications*, Springer, 2004.
- [4] W. S. Dorn, R. E. Gomory, H. J. Greenberg, Automatic design of optimal structures, *Journal de Mecanique* 3 (1964) 25–52.
- [5] K. Imai, L. A. Schmit Jr, Configuration optimization of trusses 107 (1981) 745–756.
- [6] E. A. Sadek, Dynamic optimization of framed structures with variable layout, *International Journal for Numerical Methods in Engineering* 23 (1986) 1273–1294.
- [7] M. Ohsaki, Simultaneous optimization of topology and geometry of a regular plane truss, *Computers and Structures* 66 (1998) 69–77.
- [8] W. Aichtziger, On simultaneous optimization of truss geometry and topology, *Structural and Multidisciplinary Optimization* 33 (2007) 285–304.
- [9] M. Ohsaki, *Optimization of finite dimensional structures*, CRC Press, Hoboken, NJ, 2010.
- [10] T. Hagishita, M. Ohsaki, Topology optimization of trusses by growing ground structure method, *Structural and Multidisciplinary Optimization* 37 (2009) 377–393.
- [11] J. J. McKeown, Growing optimal pin-jointed frames, *Structural optimization* 15 (1998) 92–100.
- [12] E. Ramm, K.-U. Bletzinger, R. Reiteringer, Shape optimization of shell structures, *Revue Européenne des Éléments Finis* 2 (1993) 377–398.
- [13] M. Ohsaki, T. Nakamura, M. Kohiyama, Shape optimization of a double-layer space truss described by a parametric surface, *International Journal of Space Structures* 12 (1997) 109–119.
- [14] R. McNeel, Grasshopper - algorithmic modeling for rhino, <http://www.grasshopper3d.com/>, 2018. Accessed: 2018-07-13.
- [15] R. McNeel, Rhinoceros - design, model, present, analyze, realize..., <https://www.rhino3d.com/>, 2018. Accessed: 2018-07-13.
- [16] D. Rutten, Galapagos evolutionary solver, <https://www.grasshopper3d.com/group/galapagos>, 2018. Accessed: 2018-07-13.
- [17] Rechenraum e.U., Goat - free optimization solver component for rhino's grasshopper, <https://www.rechenraum.com/en/goat.html>, 2018. Accessed: 2018-07-13.
- [18] E. Bradner, F. Iorio, M. Davis, Parameters tell the design story: Ideation and abstraction in design optimization, in: *Proceedings of the Symposium on Simulation for Architecture and Urban Design, SimAUD '14*, Society for Computer Simulation International, San Diego, CA, USA, 2014, pp. 26:1–26:8.



(a) Isometric view.



(b) Plan and elevations.

Figure 9: The solution after re-optimization of semi-spherical latticed shell ($F = 37.824[\text{N}\cdot\text{m}]$).

Table 2: Nodal locations of semi-cylindrical latticed shell [m].

| Node | Initial solution | | | Best optimal solution | | |
|------|------------------|-------|-------|-----------------------|-------|-------|
| | x | y | z | x | y | z |
| 1 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 |
| 2 | 0.250 | 0.000 | 0.188 | 0.024 | 0.015 | 0.023 |
| 3 | 0.500 | 0.000 | 0.250 | 0.294 | 0.153 | 0.207 |
| 4 | 0.750 | 0.000 | 0.188 | 0.978 | 0.015 | 0.022 |
| 5 | 1.000 | 0.000 | 0.000 | 1.000 | 0.000 | 0.000 |
| 6 | 0.000 | 0.250 | 0.000 | 0.000 | 0.295 | 0.000 |
| 7 | 0.250 | 0.250 | 0.188 | 0.041 | 0.043 | 0.039 |
| 8 | 0.500 | 0.250 | 0.250 | 0.812 | 0.135 | 0.153 |
| 9 | 0.750 | 0.250 | 0.188 | 0.958 | 0.043 | 0.040 |
| 10 | 1.000 | 0.250 | 0.000 | 1.000 | 0.367 | 0.000 |
| 11 | 0.000 | 0.500 | 0.000 | 0.000 | 0.355 | 0.000 |
| 12 | 0.250 | 0.500 | 0.188 | 0.000 | 0.489 | 0.000 |
| 13 | 0.500 | 0.500 | 0.250 | 0.502 | 0.505 | 0.250 |
| 14 | 0.750 | 0.500 | 0.188 | 0.514 | 0.506 | 0.250 |
| 15 | 1.000 | 0.500 | 0.000 | 1.000 | 0.628 | 0.000 |
| 16 | 0.000 | 0.750 | 0.000 | 0.000 | 0.635 | 0.000 |
| 17 | 0.250 | 0.750 | 0.188 | 0.042 | 0.958 | 0.040 |
| 18 | 0.500 | 0.750 | 0.250 | 0.700 | 0.799 | 0.210 |
| 19 | 0.750 | 0.750 | 0.188 | 0.951 | 0.949 | 0.047 |
| 20 | 1.000 | 0.750 | 0.000 | 1.000 | 0.690 | 0.000 |
| 21 | 0.000 | 1.000 | 0.000 | 0.000 | 1.000 | 0.000 |
| 22 | 0.250 | 1.000 | 0.188 | 0.025 | 0.983 | 0.025 |
| 23 | 0.500 | 1.000 | 0.250 | 0.973 | 0.981 | 0.027 |
| 24 | 0.750 | 1.000 | 0.188 | 0.978 | 0.985 | 0.021 |
| 25 | 1.000 | 1.000 | 0.000 | 1.000 | 1.000 | 0.000 |
| 26 | 0.250 | 0.000 | 0.288 | 0.250 | 0.000 | 0.288 |
| 27 | 0.500 | 0.000 | 0.350 | 0.500 | 0.000 | 0.350 |
| 28 | 0.750 | 0.000 | 0.288 | 0.750 | 0.000 | 0.288 |
| 29 | 0.000 | 0.250 | 0.100 | 0.000 | 0.250 | 0.100 |
| 30 | 0.250 | 0.250 | 0.288 | 0.250 | 0.250 | 0.288 |
| 31 | 0.500 | 0.250 | 0.350 | 0.500 | 0.250 | 0.350 |
| 32 | 0.750 | 0.250 | 0.288 | 0.750 | 0.250 | 0.288 |
| 33 | 1.000 | 0.250 | 0.100 | 1.000 | 0.250 | 0.100 |
| 34 | 0.000 | 0.500 | 0.100 | 0.000 | 0.500 | 0.100 |
| 35 | 0.250 | 0.500 | 0.288 | 0.250 | 0.500 | 0.288 |
| 36 | 0.500 | 0.500 | 0.350 | 0.500 | 0.500 | 0.350 |
| 37 | 0.750 | 0.500 | 0.288 | 0.750 | 0.500 | 0.288 |
| 38 | 1.000 | 0.500 | 0.100 | 1.000 | 0.500 | 0.100 |
| 39 | 0.000 | 0.750 | 0.100 | 0.000 | 0.750 | 0.100 |
| 40 | 0.250 | 0.750 | 0.288 | 0.250 | 0.750 | 0.288 |
| 41 | 0.500 | 0.750 | 0.350 | 0.500 | 0.750 | 0.350 |
| 42 | 0.750 | 0.750 | 0.288 | 0.750 | 0.750 | 0.288 |
| 43 | 1.000 | 0.750 | 0.100 | 1.000 | 0.750 | 0.100 |
| 44 | 0.250 | 1.000 | 0.288 | 0.250 | 1.000 | 0.288 |
| 45 | 0.500 | 1.000 | 0.350 | 0.500 | 1.000 | 0.350 |
| 46 | 0.750 | 1.000 | 0.288 | 0.750 | 1.000 | 0.288 |

496 [19] M. Ohsaki, K. Hayashi, Force density method for simultaneous
497 optimization of geometry and topology of trusses, *Structural*
498 *and Multidisciplinary Optimization* 56 (2017) 1157–1168.
499 [20] W. S. Hemp, *Optimum structures*, Clarendon Press Oxford,
500 1973.
501 [21] G. Farin, *Curves and Surfaces for CAGD: A Practical Guide*,
502 Morgan Kaufmann Publishers Inc., San Francisco, CA, USA,
503 5th edition, 2002.
504 [22] P. E. Gill, W. Murray, M. A. Saunders, Snopt: An sqp algorithm
505 for large-scale constrained optimization, *SIAM JOURNAL ON*
506 *OPTIMIZATION* 12 (1997) 979–1006.

Table 3: Statistical results of F [N·m] for 84 converged solutions of semi-spherical latticed shell.

| Max. | Median | Min. | Average | Std. dev. |
|--------|--------|--------|---------|-----------|
| 42.588 | 37.706 | 37.646 | 37.799 | 0.569 |

Table 4: Nodal locations of semi-spherical latticed shell [m].

| Node | Initial solution | | | Best optimal solution | | |
|------|------------------|-------|-------|-----------------------|-------|-------|
| | x | y | z | x | y | z |
| 1 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 |
| 2 | 0.250 | 0.000 | 0.188 | 0.104 | 0.000 | 0.093 |
| 3 | 0.500 | 0.000 | 0.250 | 0.158 | 0.000 | 0.133 |
| 4 | 0.750 | 0.000 | 0.188 | 0.906 | 0.000 | 0.086 |
| 5 | 1.000 | 0.000 | 0.000 | 1.000 | 0.000 | 0.000 |
| 6 | 0.000 | 0.250 | 0.188 | 0.000 | 0.085 | 0.077 |
| 7 | 0.250 | 0.250 | 0.375 | 0.163 | 0.163 | 0.273 |
| 8 | 0.500 | 0.250 | 0.438 | 0.766 | 0.177 | 0.325 |
| 9 | 0.750 | 0.250 | 0.375 | 0.845 | 0.155 | 0.261 |
| 10 | 1.000 | 0.250 | 0.188 | 1.000 | 0.082 | 0.075 |
| 11 | 0.000 | 0.500 | 0.250 | 0.177 | 0.457 | 0.394 |
| 12 | 0.250 | 0.500 | 0.438 | 0.165 | 0.168 | 0.278 |
| 13 | 0.500 | 0.500 | 0.500 | 0.503 | 0.467 | 0.499 |
| 14 | 0.750 | 0.500 | 0.438 | 0.747 | 0.480 | 0.439 |
| 15 | 1.000 | 0.500 | 0.250 | 0.824 | 0.234 | 0.325 |
| 16 | 0.000 | 0.750 | 0.188 | 0.000 | 0.892 | 0.096 |
| 17 | 0.250 | 0.750 | 0.375 | 0.154 | 0.846 | 0.261 |
| 18 | 0.500 | 0.750 | 0.438 | 0.350 | 0.760 | 0.410 |
| 19 | 0.750 | 0.750 | 0.375 | 0.837 | 0.837 | 0.273 |
| 20 | 1.000 | 0.750 | 0.188 | 1.000 | 0.908 | 0.083 |
| 21 | 0.000 | 1.000 | 0.000 | 0.000 | 1.000 | 0.000 |
| 22 | 0.250 | 1.000 | 0.188 | 0.089 | 1.000 | 0.081 |
| 23 | 0.500 | 1.000 | 0.250 | 0.603 | 0.956 | 0.282 |
| 24 | 0.750 | 1.000 | 0.188 | 0.905 | 1.000 | 0.086 |
| 25 | 1.000 | 1.000 | 0.000 | 1.000 | 1.000 | 0.000 |
| 26 | 0.250 | 0.000 | 0.288 | 0.250 | 0.000 | 0.288 |
| 27 | 0.500 | 0.000 | 0.350 | 0.500 | 0.000 | 0.350 |
| 28 | 0.750 | 0.000 | 0.288 | 0.750 | 0.000 | 0.288 |
| 29 | 0.000 | 0.250 | 0.288 | 0.000 | 0.250 | 0.288 |
| 30 | 0.250 | 0.250 | 0.475 | 0.250 | 0.250 | 0.475 |
| 31 | 0.500 | 0.250 | 0.538 | 0.500 | 0.250 | 0.538 |
| 32 | 0.750 | 0.250 | 0.475 | 0.750 | 0.250 | 0.475 |
| 33 | 1.000 | 0.250 | 0.288 | 1.000 | 0.250 | 0.288 |
| 34 | 0.000 | 0.500 | 0.350 | 0.000 | 0.500 | 0.350 |
| 35 | 0.250 | 0.500 | 0.538 | 0.250 | 0.500 | 0.538 |
| 36 | 0.500 | 0.500 | 0.600 | 0.500 | 0.500 | 0.600 |
| 37 | 0.750 | 0.500 | 0.538 | 0.750 | 0.500 | 0.538 |
| 38 | 1.000 | 0.500 | 0.350 | 1.000 | 0.500 | 0.350 |
| 39 | 0.000 | 0.750 | 0.288 | 0.000 | 0.750 | 0.288 |
| 40 | 0.250 | 0.750 | 0.475 | 0.250 | 0.750 | 0.475 |
| 41 | 0.500 | 0.750 | 0.538 | 0.500 | 0.750 | 0.538 |
| 42 | 0.750 | 0.750 | 0.475 | 0.750 | 0.750 | 0.475 |
| 43 | 1.000 | 0.750 | 0.288 | 1.000 | 0.750 | 0.288 |
| 44 | 0.250 | 1.000 | 0.288 | 0.250 | 1.000 | 0.288 |
| 45 | 0.500 | 1.000 | 0.350 | 0.500 | 1.000 | 0.350 |
| 46 | 0.750 | 1.000 | 0.288 | 0.750 | 1.000 | 0.288 |