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.)

Design and Fabrication of Structures with Graphic Statics and Augmented Reality

Norman HACK[*], Yinan XIAO[*], Carsten JANTZEN, Julian TESCHE, Karam MAWAS[a],
Markus GERKE[a], Pierluigi D'ACUNTO[bc]

[*] Technische Universität Braunschweig, Institute of Structural Design
Pockelsstraße 4, 38106 Braunschweig, Germany
n.hack@tu-braunschweig.de
yinan.xiao@tu-braunschweig.de

[a] Technische Universität Braunschweig, Institute of Geodesy and Photogrammetry
[b] Technical University of Munich, Germany; TUM School of Engineering and Design, Department of Architecture, Professorship of Structural Design
[c] Technical University of Munich, Germany; TUM Institute for Advanced Study (TUM-IAS)

Abstract

Augmented Reality (AR) has recently emerged as a powerful tool for visualizing and assembling complex spatial structures. This paper explores the transformative role of AR, extending beyond its conventional application as a 3D visualization and assembly guidance tool, to highlight the potential for the integration of structural design tools for form-finding. This innovative approach involves coupling AR with Vector-based Graphic Statics (VGS), enabling the direct and real-time manipulation of both form and force diagrams within an augmented reality environment. The paper provides a detailed account of an AR workflow applied to the construction of a 4 × 2-meter table, where AR plays a pivotal role in the structural design process. The AR interface facilitates real-time adjustments of VGS form and force diagrams, providing a dynamic and intuitive platform for designers and engineers. Additionally, AR instructions guide the precise cutting of rods, ensuring the accurate implementation of the design. Furthermore, the AR workflow encompasses orienting the 3D-printed nodes and adjusting their locations in space. The collaborative nature of the presented workflow is underscored as the entire structure takes shape in a multi-user environment using AR, showcasing its potential to enhance teamwork and efficiency in construction projects. The paper concludes by critically examining the workflow, addressing encountered bottlenecks, and proposing directions for further research. The discussion focuses on challenges in integrating VGS with AR, aiming to pave the way for innovations in the application of augmented reality in structural design.

Keywords: Augmented Reality, Vector-based Graphic Statics, form-finding, on-site construction, multi-user collaboration

1. Introduction

The introduction of Augmented Reality (AR) into the Architecture, Engineering, and Construction (AEC) sector signifies a shift towards a more immersive and interactive form of digital engagement. Overlaying digital information onto the physical world through devices such as smart glasses, mobile phones, or tablets allows users to simultaneously interact with real and virtual environments. This fusion of digital and physical realms presents new opportunities for visualizing architectural designs, engineering plans, and construction processes in real time and in their actual environments. The capacity of AR to provide immediate, contextual information transforms decision-making processes, potentially making them more informed and efficient. In the design phase, for example, AR enhances the visualization and interaction with digital models of buildings or structures before any physical construction begins. It enables architects and designers to superimpose proposed designs into real-world contexts, allowing for a deeper understanding of how a building will look and feel within its intended environment (Shin and Dunston [1]). This immersive experience facilitates better decision-making and engagement with all stakeholders by providing a tangible representation of the architectural vision.

Throughout the construction phase, AR transforms traditional methodologies by providing real-time, 3D visual guidance on-site. Construction workers can see the exact placement of elements as per the design, directly overlaid on the construction environment (Fazel and Izadi [2]). This not only increases precision and adherence to plans but also significantly enhances safety by highlighting potential hazards and preventing construction errors. Moreover, AR also offers the possibility to be linked to parametric design software, hence taking formal design decision-making directly to its physical context, generating precise instructions for collaborative manufacturing among multiple agents, including human teams (Mitterberger et al. [3], Atanasova et al. [4]) or human-robot collaborations (Mitterberger et al. [5]).

An untapped potential in the field of AR in construction is to couple the in-situ design process to a real-time structural evaluation, thus enabling structurally informed design decisions to be made directly in situ using AR. In this paper, we describe an integrated AR workflow that couples structural design methods of graphic statics with real-time AR visualization so that optimized variants of complex spatial structures can be generated by manipulating the form and force diagrams in the actual physical environment. In addition, complex connection nodes are generated from the optimized load-bearing model in an integrated downstream process, and a multi-user assembly instruction is generated. This seamless, integrated digital AR workflow, spanning from AR structural design to AR assembly, is validated using a 4 × 2-meter table as an example. This proof of concept opens up the possibility of using this workflow for larger objects in more complex environments in the future, e.g., for bridges or large-span roof structures.

2. Methods

2.1. Graphic statics as the design tool

The proposed interactive design method is based on graphic statics (GS), which is a set of geometry-based constructions that relates the structural form and the forces applied on the nodes of the structure by two reciprocal diagrams, namely the form ($\boldsymbol{F}$) and the force ($\boldsymbol{F}^*$) diagrams (Maxwell [6], Konstantatou et al. [7]). The corresponding edges in the two diagrams stay parallel, thereby maintaining the static equilibrium state in the structure (D'Acunto et al. [8], [9]). Thanks to the explicit geometric interdependency between these two diagrams, graphic statics fits itself as an intuitive technique for structural design and analysis. A computational design method and toolkit called Vector-based Graphic Statics (VGS) (Jasienski el al. [10]) is used for the integrative AR design pipeline described in this paper.

The workflow is as follows: First, an initial design (i.e., pre-design) is developed with VGS in the McNeel Rhino Grasshopper platform (McNeel et al. [11]), where the initial form and force diagrams are defined. This pre-design is then loaded into the AR environment, where it can be interactively adapted to the physical environment in situ by either modifying the force vectors in the force diagram or manually manipulating the form diagram. The equilibrium state of the designed structure is maintained during the adjustment in AR because of the interdependency between the two diagrams.

2.2. Technical basics: Data structure, Engine, Pre-design, AR workflow

Data structure: In the context of VGS, it is feasible to transform individual nodes defined by coordinates (x, y, z) in form and force diagrams. When applied to one diagram, this manipulation enforces the condition of parallelism in the other diagram. To integrate this critical condition and design constraints within an AR framework, a specific data structure and procedural workflow were devised. The foundational architecture of VGS is structured around Rhino's groupings and naming conventions. The corresponding edges maintain an identical index across both the form and force diagrams. Utilizing the rhino3dm 8.0.0 package (McNeel et al. [12]), edges from both diagrams are imported unsorted into a .NET class library, and subsequently integrated with Unity [13]. A topology identifying adjacent edges is established by sorting edges converging at a node. To ensure uniqueness, the overlayed duplicate edges in $\boldsymbol{F}^*$ in VGS are removed, creating a one-to-one correspondence between edge pairs across the form and force diagrams. Within the Unity AR environment, nodes are rendered as interactive spheres, allowing users to manipulate their positions.

Engine: An algorithm to facilitate interactive modification of form and force diagrams has been developed within the Unity AR environment. The concept is based on the explicit Dynamic Relaxation (DR) method (Barnes [14]). The method introduces a mass-dependent damping term in the motion equation, reducing transient oscillations and improving convergence to the steady-state solution. Governed by Newton's second law, the general equation to describe the motion of a single node in both diagrams can be written as (Vistnes [15]):

$$m\ddot{\boldsymbol{p}} + c\dot{\boldsymbol{p}} = \boldsymbol{f} \qquad (1)$$

where m is the mass of the node, $\boldsymbol{p}$ is the displacement vector in $\mathbb{R}^3$, $\boldsymbol{f}$ is the vector of internal nodal forces, c is the damping coefficient (set to the critical value: $2\sqrt{m}$). The superimposed dot signifies the derivative with respect to time. To transfer this second-order differential equation as a first-order differential equation, let:

$$\boldsymbol{g} = \begin{bmatrix} \dot{\boldsymbol{p}} \\ \boldsymbol{p} \end{bmatrix}, \quad \dot{\boldsymbol{g}} = \begin{bmatrix} \ddot{\boldsymbol{p}} \\ \dot{\boldsymbol{p}} \end{bmatrix} \qquad (2)$$

By substituting (2) into (1), the motion for the whole structure in either form or force diagram can be described as:

$$\mathbf{M}\dot{\mathbf{G}} + \mathbf{C}\mathbf{G} = \mathbf{F}, \qquad (3)$$

with:

$$\mathbf{M} = \begin{bmatrix} m\mathbf{I}_3 & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{I}_3 \end{bmatrix}, \ \mathbf{C} = \begin{bmatrix} c\mathbf{I}_3 & \mathbf{0}_{3\times3} \\ (-1)\mathbf{I}_3 & \mathbf{0}_{3\times3} \end{bmatrix}, \ \mathbf{G} = [\boldsymbol{g}_1 \ \cdots \ \boldsymbol{g}_n], \ \mathbf{F} = \begin{bmatrix} \boldsymbol{f}_1 & \cdots & \boldsymbol{f}_n \\ \mathbf{0}_{3\times1} & \cdots & \mathbf{0}_{3\times1} \end{bmatrix}$$

solve (3), it derives:

$$\dot{\mathbf{G}} = \mathbf{M}^{-1}(\mathbf{F} - \mathbf{C}\mathbf{G}) \qquad (4)$$

discretize (4) as:

$$\mathbf{G}_{t+1} = \mathbf{G}_t + \Delta t \cdot \mathbf{M}^{-1}(\mathbf{F} - \mathbf{C}\mathbf{G}_t) \qquad (5)$$

where Δt is a fixed time increment. The Runge-Kutta 4th Order (RK4) method (Dormand and Prince [16]) has been implemented to solve this differential equation for a highly precisive numerical solution until the difference between $\mathbf{G}_{t+1}$ and $\mathbf{G}_t$ is less than the tolerance value. So far, the only unknown element in (5) is $\mathbf{F}$, which is the force to maintain the parallelization between the corresponding edges in form and force diagrams. Since the application processes on both diagrams are similar, it is enough to introduce the iterative process considering only the force diagram while the form diagram is fixed. For each iteration, the force of the node V_i^* in the force diagram is updated as:

$$\boldsymbol{f}_i = \frac{\sum_{j\in E_i}(\|\boldsymbol{e}_{ij}^*\| * \boldsymbol{r}_{ij})}{2n(E_i)} \qquad (6)$$

$$\boldsymbol{r}_{ij} = sign * (\hat{\boldsymbol{e}}_{ij} - (\hat{\boldsymbol{e}}_{ij}\hat{\boldsymbol{e}}_{ij}^*) * \hat{\boldsymbol{e}}_{ij}^*) \qquad (7)$$

in which E_i stands for all the connected lines to node V_i^*, $\boldsymbol{e}_{ij}^*$ and $\boldsymbol{e}_{ij}$ are the vector of the edges in the force diagram and the corresponding ones in the form diagram. The superimposed hat represents the normalized vector. $\boldsymbol{r}_{ij}$ is the residual vector describing the parallelization between the paired edges in form and force diagrams. Respecting the framework of VGS, in the force diagram it may appear duplicated edges representing the same element but with opposite directions. The $sign$ here is to maintain the opposite directions of the duplicated edges which is set as $\frac{\hat{e}_{ij}\hat{e}_{ij}^*}{\|\hat{e}_{ij}\hat{e}_{ij}^*\|}$ (Figure 1).

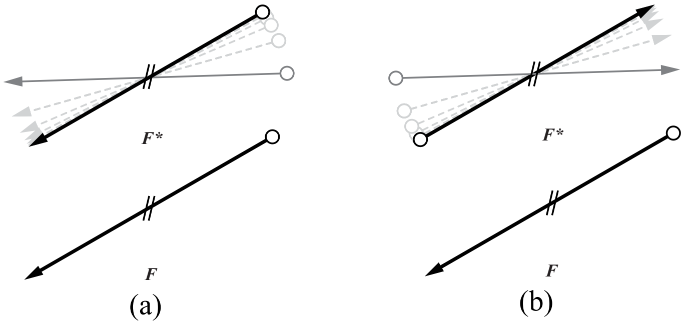


Figure 1: The algorithm of parallelization maintains the opposite directions of the duplicated edges in $\boldsymbol{F}^*$, corresponding to the same edge element in $\boldsymbol{F}$.

Pre-design: Based on the principles of graphic statics described above, the generic pre-design of a table was developed, which can later be adapted in AR to meet the specific conditions of the environment. The objective of the structural design is to achieve a 4-meter-span tabletop while also meeting various architectural requirements, such as adhering to the prescribed table height limitations and ensuring sufficient space underneath for use. The initial prototype is two parallel arches under designated uniform loads $150N/m^2$, and the table's height is fixed $750mm$. The strut elements are designed with aluminum rods with hollow circular cross-sections, featuring a thickness of 1mm. In order to span the 4-meter gap, the table's top exceeds the desired target by $145mm$ (Figure 2a). Besides, the space beneath the arch is limited for use, especially the area around the supports. To address these challenges, the first step is to invert underneath the protruding portion on the top (Figure 2b) to fulfil the flat height restriction of the table. Subsequently, the sub-structures around the support area are replaced to create more usable space by integrating tension elements (Figure 2c). Finally, the distribution of the uniform loads are adjusted to be proportional to the corresponding area of the applied nodes (Figure 2d). The design for the side view of the structure starts from a flipped arch-cable (Figure 2e). This approach involves elevating the base while simultaneously reducing the width between the support nodes, thus creating usable space around the table legs (Figure 2f - 2g). To further enhance stability, the directions of the reaction forces at the supports are redirected by introducing tension-bearing elements (Figure 2h). The completed design, illustrated below, is capable of supporting a tabletop measuring 4 × 1.8 meters (Figure 3).

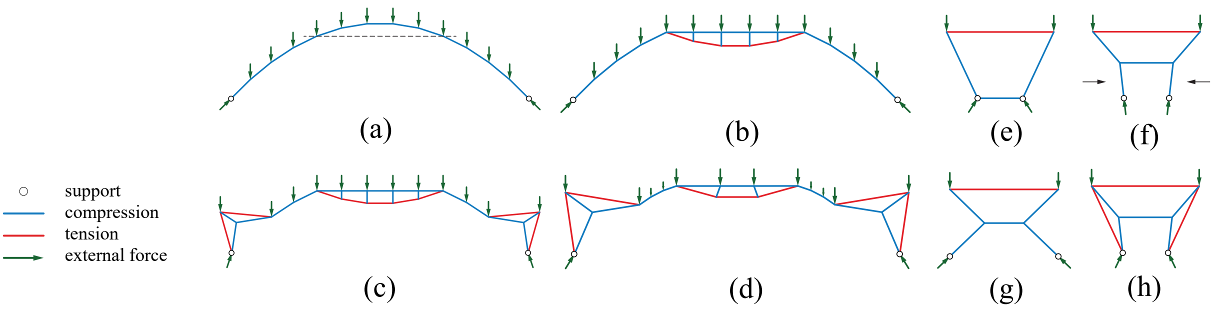


Figure 1: Design of a table: (a-d) the procedural steps involved in designing the front; (e-h) the procedural steps involved in designing the side view.

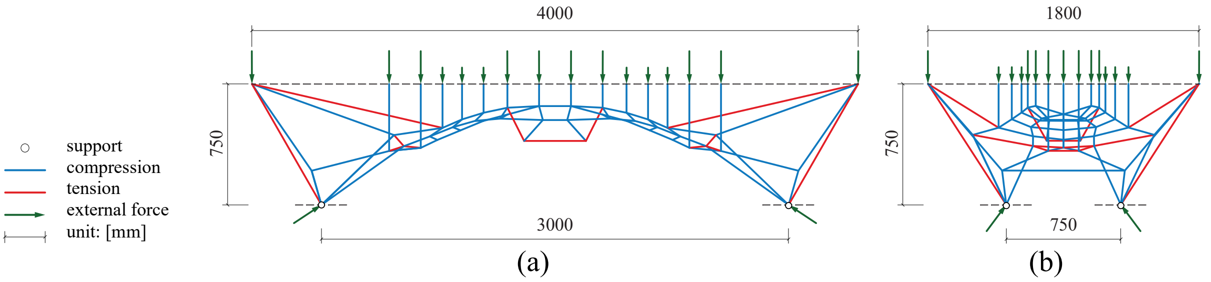
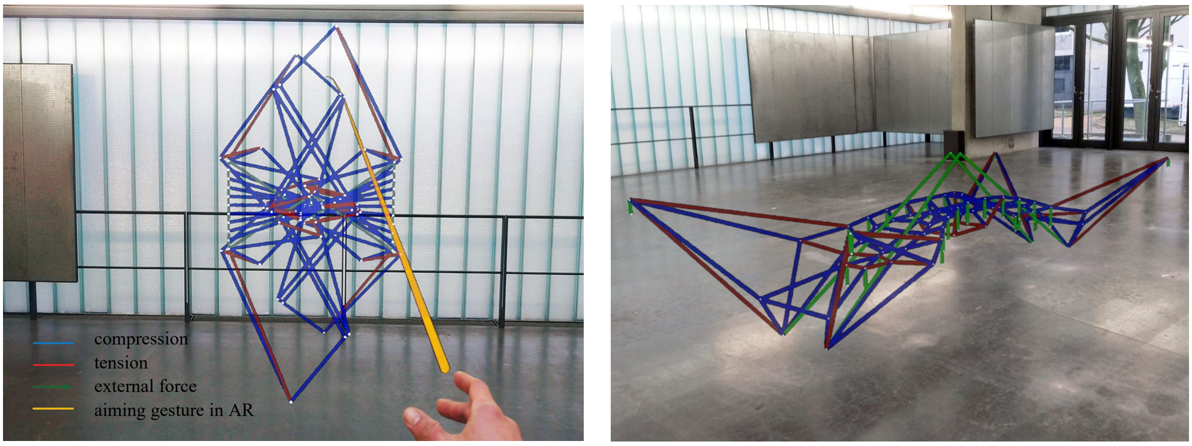


Figure 2: The front (a) and the left (b) view of the completed design. Integrating the tension-bearing elements allows shrinking the table supports and reversing the middle part of the original arch structure to achieve the designated spanning distance and height limitations of the table, and to leave enough space underneath to be used.

AR workflow: Using the Microsoft HoloLens 2 [17] device, the Augmented Reality (AR) workflow within the context of a VGS application can be outlined as follows: After loading the pre-design, a user engages with the system by manipulating the spheres within the AR environment, typically through a gesture or touch. This action triggers the repositioning of the endpoints of edges that are connected to the same sphere, effectively altering their spatial configuration (Figure 4). Subsequently, the system identifies and selects associated pairs of edges that correspond across the form and force diagrams. If the adjustment of any edge fails to meet the equilibrium condition in the structure, the engine solver introduced above is activated to achieve a stable result. Additionally, several constraints can be integrated during the manual modification, such as the fixed support location and the direction or magnitude of inner force vectors in specific edge elements.



(a) (b)

Figure 3: Designing with the force (a) and form (b) diagram in the AR Environment: Moving a node in the F (or F*) using gesture defined in Microsoft Hololens2, the other diagram F* (or F) automatically adjusts to align the corresponding edges in parallel between both diagrams.

2.3 Joint design and 3D printing

After adapting the pre-design in the AR environment, the next critical step is the materialization of the strut-and-tie form diagram. This comprehensive process entails the detailed dimensioning of both compression and tension edge members, along with the automated fabrication of the joints (Oztoprak et al. [18]). Both the form and force diagrams are saved as a CAD file. This file is then opened within the McNeel Rhino Grasshopper environment for further processing. Using the strut-and-tie model, joints were created based on the Sub-D geometry type with the Rhino command "Multipipe." In scenarios where forces exerted upon members exceed predefined thresholds, a strategy of duplicating members is employed instead of scaling, ensuring structural integrity and resilience (Xiao et al. [19], Tanadini et al. [20]). Furthermore, each component is marked with a unique identifier (ID) embedded within a discrete, non-visible section of the structure.

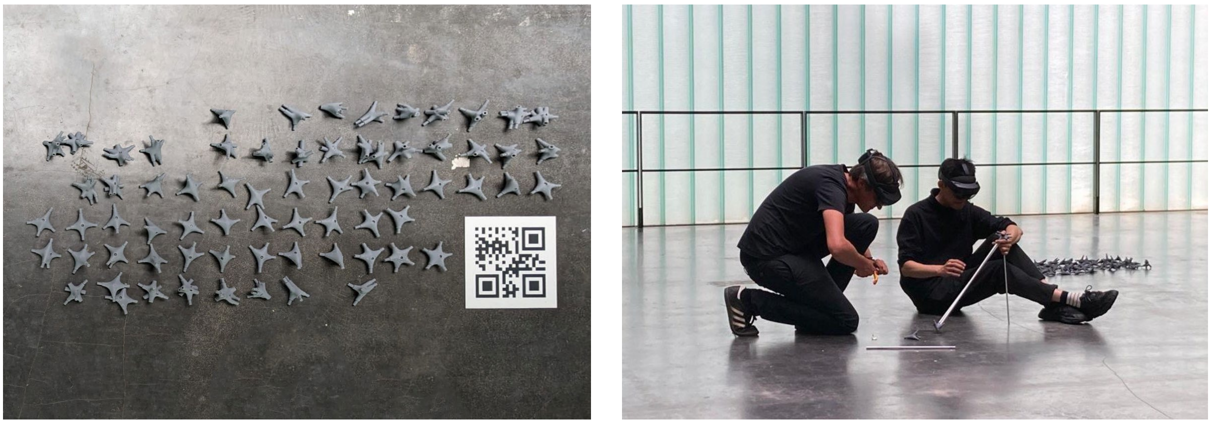
The joints are exported as STL files and printed with SLA resin. The construction between the joints and the compression rod elements incorporate a plug-in mechanism. In contrast, tension connections utilize a combination of push-through and screw techniques for assembly (Figure 5).

2.4 Assembly method, AR setup, voice control

Upon automatic generation of joints from the 3D model for fabrication, the assembly process unfolded as follows: The materials selected for the rods included 8 mm diameter aluminum rods with a 1 mm wall thickness for compression elements and 6 mm diameter rods for tension elements. An assembly setup was established by generating and printing a QR code, which acted as the origin point (0,0,0) for registration, defining the local coordinate system for the assembly setup (Figure 6a). The printed joints were then arranged in a virtual grid within a spacious assembly area. The tension and compression rods were cut on demand in situ instead of pre-prepared.

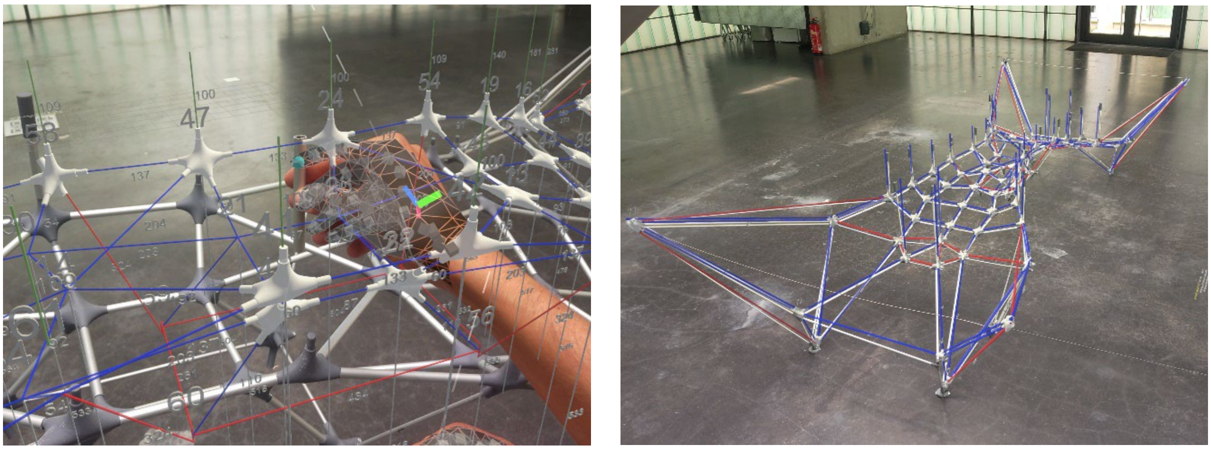
(a) (b)

Figure 4: Additive manufacturing of the generated joints: (a) connections to the compression rod elements incorporate a plug-in mechanism; (b)) connections to the tension rod elements utilize a combination of push-through and screw strategy.



(a) (b)

Figure 5: In-situ assembly with HoloLens 2: (a) QR code as origin and virtual grid of joints; (b) Two people cooperate in the AR environment simultaneously. One person aligns with the virtual structure while the other serves as an assembler.



(a) (b)

Figure 6: (a) The joints and rods are placed according to the 3D Assembly plan; (b) The final assembled substructure with the AR overlay of the form diagram in VGS.

The assembly process involved two individuals, designated A and B, working in tandem while wearing AR glasses (Figure 6b). Person A serves as the assembler, and Person B assists by handing over joints and supporting the structure. A third participant, person C, was responsible for cutting the rod elements.

Throughout the assembly, A and B were able to view the centerline model in AR, including rod types and lengths, while commands for displaying joint positions could be issued via voice control (Figure 7). This approach facilitated the efficient identification and assembly of joints and rods. The entire structure was assembled through a collaborative effort, with a constant comparison of construction status via AR overlay, allowing for immediate error detection and correction.

Figure 8: Final assembled table substructure with virtual tabletop.

3. Discussion

In the context of in-situ AR design applications, performance problems occur when complex structures with 20 or more connections are involved. The primary challenge encountered is a significant delay in reaching equilibrium, attributed to slow data transfer to the display and computational limitations. Despite static calculations being processed on a PC rather than directly on the Augmented Reality glasses, these bottlenecks affect system efficiency.

Several improvements are proposed to enhance performance and usability in future iterations. Key among these is the numerical stabilization of calculations within the RK4 method, potentially involving an increase in precision by shifting from float to double data types and implementing dynamic step size control. Additionally, code refactoring is recommended to streamline operations. The inherent limitation of AR glasses in natively reading Rhino files necessitates running applications in Unity's debug mode, resulting from the inability to compile Rhino libraries and dependencies for the HoloLens 2 with Windows 10 and ARM architecture. Establishing a separate exchange format for data could mitigate this issue, facilitating faster software setup.

4. Conclusion and future work

With a focus on in-situ structure design, this study presented a comprehensive digital workflow for designing, calculating, visualizing, and constructing complex spatial structures using Augmented

Reality. This integration of structural consideration in the design phase marks a significant advancement in applying AR technologies within the Architecture, Engineering, and Construction sectors. While the current case study focused on a straightforward example of designing a table, future investigations will extend to more complex and spatially challenging projects.

Further research directions include enhancing user interaction capabilities within the AR glasses to provide accessible functions such as strain energy visualization for recognizing equilibrium state completion. The dimensioning of joints could also be automated based on force vectors in the force diagram, optimizing joint shape and size. Potential future applications could include intricate construction scenarios where closer physical constraints necessitate an in-situ design approach. This would allow for a more immediate and context-sensitive response to environmental constraints.

Acknowledgments

The research has been conducted within the scope of the research project "Integrating Additive Manufacturing in the Construction Process" (C06) within the collaborative research center "Additive Manufacturing in Construction - The Challenge of Large Scale," funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) - project number 414265976 - TRR 277, as well as by the Matthaei - Stiftung.

References

[1] D. H. Shin and P. S. Dunston, "Identification of application areas for Augmented Reality in industrial construction based on technology suitability," *Autom Constr*, vol. 17, no. 7, pp. 882–894, Oct. 2008, doi: https://doi.org/10.1016/j.autcon.2008.02.012.

[2] A. Fazel and A. Izadi, "An interactive augmented reality tool for constructing free-form modular surfaces," *Autom Constr*, vol. 85, pp. 135–145, Jan. 2018, doi: https://doi.org/10.1016/j.autcon.2017.10.015.

[3] D. Mitterberger, E. M. Angelaki, F. Salveridou, R. Rust, L. Vasey, F. Gramazio and M. Kohler, "Extended reality collaboration: Virtual and mixed reality system for collaborative design and holographic-assisted on-site fabrication," in *Towards Radical Regeneration*, C. Gengnagel, O. Baverel, G. Betti, M. Popescu, M. R. Thomsen, and J. Wurm, Eds., Cham: Springer International Publishing, 2023, pp. 283–295, doi: https://doi.org/10.1007/978-3-031-13249-0_24.

[4] L. Atanasova, B. Saral, E. Krakovská, J. Schmuck, S. Dietrich, F. Furrer, T. Sandy, P. D'Acunto and K. Dörfler, "Collective AR-assisted assembly of interlocking structures," in *Towards Radical Regeneration*, C. Gengnagel, O. Baverel, G. Betti, M. Popescu, M. R. Thomsen, and J. Wurm, Eds., Cham: Springer International Publishing, pp. 175–187, 2023, doi: https://doi.org/10.1007/978-3-031-13249-0_15.

[5] D. Mitterberger, L. Atanasova, K. Dörfler, F. Gramazio and M. Kohler, "Tie a knot: human–robot cooperative workflow for assembling wooden structures using rope joints," *Construction Robotics*, vol. 6, no. 3, pp. 277–292, 2022, doi: https://doi.org/10.1007/s41693-022-00083-2.

[6] J. C. Maxwell, "On reciprocal figures, frames, and diagrams of forces," *Earth Environ Sci Trans R Soc Edinb*, vol. 26, no. (1), pp. 1–40, 1870.

[7] M. Konstantatou, P. D'Acunto, and A. McRobie, "Polarities in structural analysis and design: n-dimensional graphic statics and structural transformations," *Int J Solids Struct*, vol. 152–153, pp. 272–293, 2018, doi: https://doi.org/10.1016/j.ijsolstr.2018.07.003.

[8] P. D'Acunto, J. P. Jasienski, P. O. Ohlbrock, C. Fivet, J. Schwartz and D. Zastavni, "Vector-based 3D graphic statics: A framework for the design of spatial structures based on the relation between form and forces," *Int J Solids Struct*, vol. 167, pp. 58–70, Aug. 2019, doi: https://doi.org/10.1016/j.ijsolstr.2019.02.008.

[9] P. D'Acunto, J.-P. Jasienski, P. O. Ohlbrock and C. Fivet, "Vector-Based 3D Graphic Statics: Transformations of Force Diagrams," in *Proceedings of IASS Annual Symposium*, A. Bögle and M. Grohmann, Eds., Hamburg, Germany: Proceedings of the IASS Annual Symposium, 2017.

[10] J.-P. Jasienski, Y. Shen, P. O. Ohlbrock, D. Zastavni and P. D'Acunto, "A computational implementation of Vector-based 3D Graphic Statics (VGS) for interactive and real-time structural design," *Computer-Aided Design*, p. 103695, Feb. 2024, doi: https://doi.org/10.1016/j.cad.2024.103695.

[11] Robert McNeel & Associates, "Rhino - Rhinoceros 3D." Accessed: Apr. 09, 2024. [Online]. Available: https://www.rhino3d.com/.

[12] Robert McNeel & Associates, "Libraries based on OpenNURBS with a RhinoCommon style." Accessed: Apr. 09, 2024. [Online]. Available: https://github.com/mcneel/rhino3dm.

[13] Unity Technologies, "Unity Real-Time Development Platform | 3D, 2D, VR & AR Engine." Accessed: Apr. 09, 2024. [Online]. Available: https://unity.com/.

[14] M. R. Barnes, "Form-finding and analysis of prestressed nets and membranes," *Comput Struct*, vol. 30, no. 3, pp. 685–695, Jan. 1988, doi: https://doi.org/10.1016/0045-7949(88)90304-5.

[15] A. I. Vistnes, "Numerical Methods," in *Physics of Oscillations and Waves: With use of Matlab and Python*, A. I. Vistnes, Ed., Cham: Springer International Publishing, pp. 59–91, 2018. doi: https://doi.org/10.1007/978-3-319-72314-3_4.

[16] J. R. Dormand and P. J. Prince, "A family of embedded Runge-Kutta formulae," *J Comput Appl Math*, vol. 6, no. 1, pp. 19–26, Mar. 1980, doi: https://doi.org/10.1016/0771-050X(80)90013-3.

[17] Microsoft, "HoloLens 2—Overview, Features, and Specs | Microsoft HoloLens." Accessed: Apr. 09, 2024. [Online]. Available: https://www.microsoft.com/en-us/hololens/hardware#document-experiences.

[18] O. Oztoprak, A. Paolini, P. D'Acunto, E. Rank and S. Kollmannsberger, "Two-scale analysis of spaceframes with complex additive manufactured nodes," in *Engineering Structures*, vol. 289, p. 116283, 2023, doi: https://doi.org/10.1016/j.engstruct.2023.116283.

[19] Y. Xiao, Y. Shen, N. Hack and P. D. Acunto, "A structure- and fabrication-informed strategy for the design of lattice structures with Injection 3D Concrete Printing," in *IASS/APCS 2022 Beijing Symposium: Next Generation Parametric Design*, Beijing: International Association for Shell and Spatial Structures (IASS), pp. 1-12 (12), Sep. 2022.

[20] D. Tanadini, P. O. Ohlbrock, M. Kladeftira, M. Leschok, E. Skevaki, B. Dillenburger and P. D'Acunto, "Exploring the potential of equilibrium-based methods in additive manufacturing: the Digital Bamboo Pavilion," *in IASS/APCS 2022 Beijing Symposium: Next Generation Parametric Design*, Beijing: International Association for Shell and Spatial Structures (IASS), pp. 1-12 (12), Sep. 2022.