

Emergent Robotic Assembly Design

Embedding Structural Feedback into Agent-Based Modelling for Collective Robotic Construction

Samuel Leder¹ and Achim Menges²

^{1,2}*Institute for Computational Design and Construction (ICD), & Integrative Computational Design and Construction for Architecture (IntCDC), University of Stuttgart*

Intelligent Construction and Building Systems Group, Max Planck Institute for Intelligent Systems

¹*samuel.leder@icd.uni-stuttgart.de, 0000-0003-3533-3633*

Abstract. As collective robotic construction (CRC) systems become technologically advanced, structural stability is an increasingly important research theme. However, structural analysis in CRC is challenging when analysing the local effects of many robots adding building elements to the same structure. This paper introduces an agent-based modelling and simulation (ABMS) approach to emergent CRC that embeds deflection-aware structural modelling into digital assembly processes. It is explored through a group of robots executing local behaviors, guided by an ϵ -greedy policy for locomotion and building element placement, to assemble cantilever structures within a guide volume toward a target direction without any predefined blueprint. Each proposed element is added to a finite element analysis (FEA) model and evaluated for maximum deflection under predefined loading and robot weight; any placement exceeding a deflection threshold is rejected. Simulations varying agent model parameters assessed success rate, maximum deflection, total iterations, total elements placed, cantilever length and highest level reached. Results indicate that increasing swarm size markedly improves success rate and cantilever reach. This preliminary study demonstrates that tightly coupling agent decision-making with structural evaluation enables self-organizing assemblies that maintain structural performance criteria throughout the assembly process.

Keywords. Collective Robotic Construction, Agent-Based Modelling, Emergence, Finite Element Analysis (FEA), Mobile Robotics

1. Introduction

Collective robotic construction (CRC) has emerged as an approach for construction automation, leveraging large teams of mobile robots to autonomously assemble large-

scale structures (Petersen et al., 2019). By distributing tasks among many agents, CRC has the potential to achieve robustness to individual failures, scalability across project sizes, accelerated build rates, and adaptability to unstructured environments through massively parallel operations. However, coordinating large numbers of robots simultaneously presents a significant structural engineering challenge: the incremental, decentralized addition of building elements creates a globally interdependent building system in which local placement decisions can critically affect the stability of an entire assembly.

To date, research on CRC has largely avoided this problem by assembling simple structures or focusing on other research topics, such as heterogeneous robotic systems (Yablonina, Kalousdian and Menges, 2020), decentralized control algorithms (Yun, Schwager, and Rus, 2011) and methods for robustness to error (Chen and Petersen, 2022). However, as CRC research is beginning to transition from simulations to desktop showcases and more recently to large-scale demonstrations, structural performance will become a more critical factor to consider. Continuous structural assessment becomes essential to prevent failures and ensure safe, reliable performance.

This need is amplified in emergent CRC systems, where no global blueprint exists and the final geometry evolves through local interactions rather than predefined designs with global planners. While self-organization confers adaptability and resilience, it also renders global structural performance unpredictable: small, local additions can propagate instabilities throughout the structure. In contrast to predefined designs, where structural calculations and build sequences are determined offline, emergent CRC requires feedback that ties each agent's local action to its global load-bearing consequences. Decentralization is particularly advantageous under uncertainty in environment, assembly state, or sequencing, where precomputed plans become brittle and in-situ adaptation is required.

To address the challenge of stability in emergent CRC, this paper introduces an agent-based modelling and simulation (ABMS) approach that embeds deflection-aware structural evaluation directly into the emergent assembly loop. In this framework, structural behaviour is not a post-rational analysis but an active, generative constraint that shapes emergent form during construction. The research is grounded in a single typology: a cantilevered span grown within a guide volume toward a fixed target. The model is based on a CRC system developed in prior work (Leder et al. 2022). The paper discusses the agent-based model (ABM) as well as a set of simulations conducted with the model. The primary contribution of this paper is a structurally conditioned, deflection-aware ABMS approach, which includes a tight integration of decentralized agent behaviors with incremental, per-placement structural evaluation.

This study intentionally focuses on a single assembly typology as a minimal testbed for examining how decentralized agents can be coupled with in-loop structural evaluation. As such, the contribution is a reproducible mechanism for embedding structural feedback into agent decision-making without a predefined blueprint. Deflection is employed as a computationally tractable global constraint evaluated incrementally during assembly; other performance criteria (e.g., stress, buckling, or connection capacity) are excluded at this stage to isolate system dynamics. The resulting experiments are exploratory but systematic, aimed at revealing scaling

behaviour, parameter sensitivity, and inherent limitations of emergent construction.

2. Background

CRC investigates how large teams of simple, autonomous robots can assemble structures larger than any individual robot. A broad range of platforms has explored different materials, kinematics, and coordination strategies. Reviews by Petersen et al. and Ardiny et al. summarize CRC's advantages over single-machine automation while highlighting the technical interdependencies that arise when scaling to swarms (Petersen et al., 2019; Ardiny, Witwicki, and Mondada, 2015). Two lines of prior work in the larger field are especially relevant for this research: stability mechanisms during construction and ABMS as a control and generative design framework.

In conventional construction, structural integrity is typically verified only for the final state of a building. Failures during construction are mitigated by extensive scaffolding, formwork, and redundant support systems, rendering mid-construction instability a minor concern (Melenbrink, Werfel, and Menges, 2020). CRC aims to reduce reliance on such global supports, so each placement immediately alters load paths and can create global weak points, motivating in-loop structural feedback (Petersen et al., 2019). Recent efforts used local force sensing to build unsupported cantilevers with external manipulators (Melenbrink and Werfel, 2017) and with self-assembling modules with on-board sensors (Bray and Groß, 2021; Swissler and Rubenstein, 2022); and Hosmer et al. integrated a learned structural predictor (graph neural network) to recommend stabilizing connections in real time (Hosmer, 2024). These approaches either operate primarily in two dimensions, rely on specialized hardware, or trade generality for guarantees. To balance fidelity and scalability, this paper adopts a per-placement deflection check as a general structural signal during assembly.

ABMS has been used in architecture and construction to study how local interactions yield global patterns spanning form-finding, crowd dynamics, and urban planning (Stieler et al., 2022). It has also been used to embed structural information into generative design processes (Udaykumar et al., 2023). Within CRC specifically, ABMS has generated offline blueprints that swarms later realize physically (Kayser et al., 2018; Pietri and Erioli, 2017) and served as a cyber-physical control layer that mirrors robots in real time for continuous feedback and adaptive coordination (Leder et al., 2024b). Building on these precedents, this paper integrates structural evaluation directly into the agent decision cycle so that emergent growth is constrained by structural performance at each step, without reliance on predefined blueprints or external supports.

3. Methodology

3.1. ASSUMPTIONS

A swarm of climbing robots that assemble linear building elements with point connections between their long faces was considered. This is based on the CRC system in Figure 1 being developed as part of the Integrative Computational Design and Construction for Architecture (IntCDC) (Leder et al. 2022, 2024c).

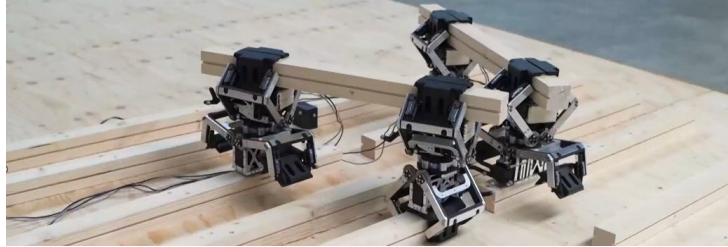


Figure 1. CRC system of robotic actuators that form kinematic chains together with linear timber building elements and utilize the building elements to assemble structures as well

In the ABM for this paper, each agent represents a single climbing robot. Locomotion occurs via rotation around the agent's current position, enabling motion between linear building elements in the structure being built. To simplify and accelerate experiments, each agent independently carries one building element, no multi-robot transport is required, and all results are simulation-only. Simulation is utilized at this stage to: (i) safely explore parameter ranges and (ii) measure structural effects of local behaviors without the complexities of physical hardware.

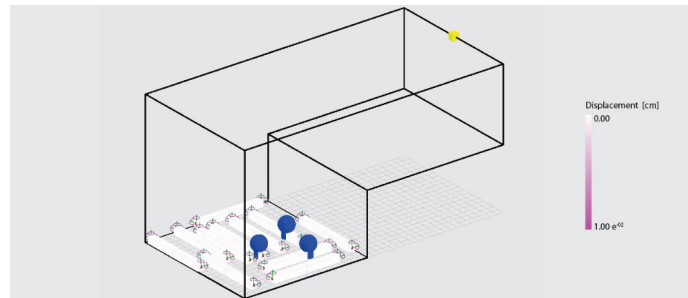


Figure 2. Setup configuration for the ABM to run. Robots are in blue. The guide volume is the black line. The target is the yellow cube. The building elements are depicted with a gradient indicating the displacement in millimetres. Support points are marked with coordinate axes.

All experiments target a cantilevered span that grows within a rectangular guide volume toward a fixed target (Figure 2). The task is to reach the target while keeping incremental deflection under a structurally informed limit discussed in Section 3.3. Trials start with agents randomly placed on a first layer of pre-assembled elements whose endpoints are rigidly supported. New elements attach to the element they contact below to ensure connectivity. If agents place elements into the ground layer, those are assumed rigidly supported at both ends. After placing an element, an agent reloads by returning toward the origin; a distance threshold allows reload without requiring the agent to reach the exact origin.

While robots cannot directly “measure deflection,” the same decision logic can be deployed on hardware by using local sensing to maintain a compact digital map of the evolving structure and enforcing the same check; alternatively, without a full digital

model, the ABM can define deployable behaviors whose accept/reject rule is triggered by a calibrated deflection proxy obtained by fusing onboard sensors

3.2. ABM

This section describes the agent–environment interaction workflow, from candidate generation to structural validation.

While reinforcement learning (RL) can, in principle, optimize a cantilevering task with a well-defined reward, this study adopts an ABMS framework to reveal how decentralized feedback shapes emergent growth. The objective here is not to learn a fixed policy but to study the link between simple local rules and global stability in an explicit and reproducible. ABMS provides lightweight, interpretable control over decision parameters, enabling systematic experiments without the additional complexity of RL. In this way, ABMS serves as a transparent substrate for understanding agent–structure coupling; future work can layer RL atop to adapt exploration schedules or derive placement heuristics once these dynamics are characterized. Although demonstrated here for a cantilever construction case, the workflow is abstracted around three transferable components, local candidate generation, decentralized exploration, and a global structural veto, which can be reparametrized for other materials, element typologies, or performance criteria.

The ABM formalization was implemented as an extension to ABxM, an open-source, object-oriented agent framework for Grasshopper/Rhino (L. Nguyen et al., 2022). ABxM allows users to define new agent and environment classes via C# scripts or visual-programming components. Two agent systems were utilized to represent the physical system: (i) a matrix representation of the environment, storing information on the built structure, and (ii) a cartesian representation of the robots, storing information on the state of the climbing robots. Customization of the core class from ABxM using custom dictionaries was required. Additionally, a new agent system C# class, the Karamba Matrix Environment, was built on top of the Matrix3dAgentSystem to allow for the execution of the finite element analysis (FEA).

3.2.1. Karamba Matrix Environment

The Karamba Matrix Environment stores the evolving state of the built structure in a voxel grid of size $U \times V \times W$ with cell size that matches the linear building element's cross section. Representing the environmental state as matrices enables constant-time neighbourhood queries instead of expensive geometric intersection tests. The environment maintains matrices representing the current assembly state, including a *BuildingElementMatrix* that encodes the location and occupancy of placed building elements, a *BuildingZoneMatrix* that defines the permissible build volume constraining candidate placements, and directional scalar fields (*LoadingDirectionMatrix* and *TargetDirectionMatrix*) that guide agent movement toward supply and target regions. A list of proposed building-element geometries (*NextBuildingElementCurves*) is generated each iteration and evaluated for placement. These matrices are initialised at the start of each simulation and updated incrementally after every accepted placement.

3.2.2. Cartesian Agent System

The Cartesian Agent System is utilized to model the climbing robots as discrete agents with state variables indicating whether the agent currently is loaded and waiting to determine if its placement is successful. Three agent behaviors drive the agent actions at each iteration of the model:

- **Loading Behaviour** checks an agent's distance to the supply origin. If within the reload threshold and not already loaded with a building element, it reloads the agent.
- **Movement Behaviour** implements an ϵ -greedy policy utilizing an exploration–exploitation decay rate to move the agents around the structure depending on their loaded state. An input parameter determines the radius of the circle on where the agent can move. With probability ϵ , the agent selects a random reachable point to locomote; otherwise with $1 - \epsilon$, it moves in the direction that maximizes a scalar field, *LoadingDirectionMatrix* when not loaded with a building element, *TargetDirectionMatrix* when loaded, guiding agents back to supply or in the target direction.
- **Placement Behaviour** is executed when an agent is loaded and follows an ϵ -greedy policy. Adjacent empty voxels are sampled as placement candidates with orientations selected from a predefined rotation set. Candidates must lie within the build volume, connect to the existing structure, avoid intersections, and satisfy local stacking constraints. With probability ϵ , candidates are selected randomly; otherwise, selection follows the *TargetDirectionMatrix*. Each valid candidate triggers the incremental structural evaluation described in Section 3.3.

3.3. INCREMENTAL STRUCTURAL EVALUATION AS A GLOBAL CONSTRAINT

While deflection is the only explicitly evaluated structural metric, it functions as a global constraint that implicitly integrates geometry, material stiffness, span length, cumulative loading, and construction sequence at each placement decision.

Structural feedback is provided by translating the environment information into a FEA model utilizing Karamba inside of Grasshopper (Preisinger, 2013). When an agent proposes a new building element, its corresponding Rhino curve together with the curves already in the structure are assembled into a Karamba model of beam elements. The physical properties of the building elements are taken from the physical CRC system that inspired this research (Leder et al. 2022). All elements share uniform wooden properties ($E = 10$ GPa, density = 600 kg/m^3), selected as representative values for structural timber, and a square cross-section equal to the voxel size ($50\text{mm} \times 50\text{mm}$), with pinned endpoint connections simulating metal screw fasteners. Gravity loading is applied to every element, and a concentrated downward point load (approximately 10 N) at the candidate building element's longest cantilevering endpoint represents the robot's weight during placement. Supports are pinned at both ends of any first-layer building element, including newly placed ones. Karamba3D's static linear-elastic solver is then invoked under default tolerances; the maximum nodal deflection is extracted and compared to a scale-consistent serviceability limit. Specifically, a placement is accepted only if $\delta_{\max} \leq \alpha L_{\text{ref}}$ with $\alpha = 1/250$, where

L_{ref} is the current cantilever span (distance from the nearest support to the tip) before accepting the candidate. If this $L/250$ limit is exceeded, the placement is rejected and the candidate is removed; otherwise, the building element is committed to the environment. Since the agents are also unaware of the placement suggestions of each other, reject may also occur because two agents attempt to place building elements that overlap in this stage, overlaps are deterministically detected and refused prior to commitment.

4. Results

The ABM was evaluated through systematic parameter sweeps to quantify system-level trends in success probability, structural reach, convergence speed, and deformation margins under controlled conditions (Table 1). An initial test of the exploration–exploitation decay rate (decay rate $\in \{0.5, 0.1, 0.01\}$) with three agents showed that a decay rate of 0.5 produced the most successful performance in terms of reaching the target and was therefore used for the other agent counts. The evolution of one trial with three agents is depicted in Figure 3. An additional sweep explored different rotation options (Rotation Options = 1, 5, 10) for the three agent case. For each parameter combination, five independent trials were run for up to 1000 iterations, terminating early if no valid placements were found for 60 consecutive iterations or if a single agent reached the target. Reported values represent the mean of five trials. The maximum allowable deflection was set to 2mm based on a 0.5m building-element length. Recorded metrics per trial included success rate, maximum deflection at the final accepted placement (mm), total model iterations until success or termination, total elements placed, cantilever length at termination or success (m, measured from support to tip) and highest level reached.

Table 1. Metrics recorded from the simulations with different agent counts, decay rates and rotation options for the Placement Behaviour

Agent Count	Decay Rate	Rotations	Success Rate (%)	Maximum Deflection (mm)	Total Iterations	Total Elements	Cantilever Length (m)	Level Reached
3	0.001	1	0	0.268	424.80	59.2	0.2458452	9.4
3	0.1	1	20	0.489	567.00	77.4	0.550005	12.8
3	0.5	1	40	0.374	590.00	79.2	0.5201	12.4
3	0.5	5	0	0.152	396.60	51.6	0.23654	9.4
3	0.5	10	0	0.018	253.00	45	0.1474168	6.8
5	0.5	1	60	0.729	461.80	79.8	0.66	14.6
7	0.5	1	40	0.757	451.40	80	0.580105	14.2
10	0.5	1	80	0.894	387.40	93.8	0.79	17
15	0.5	1	100	1.691	299.20	98.8	1.0502	19

Table 1 summarizes the results. Swarm size emerged as a primary scaling variable, producing monotonic improvements in success rate, cantilever reach, and convergence

speed while maintaining deflection within serviceability limits. Increasing the number of agents led to consistently higher success rates, greater structural reach, and shorter simulation times. Success rate rose from 40% at 3 agents to 100% at 15 agents, while the corresponding cantilever length increased from 0.52m to over 1.05m. The total number of placed elements and levels reached also grew proportionally, from roughly 80 elements and 12 levels for small swarms to nearly 100 elements and 19 levels at fifteen agents. Meanwhile, total iterations decreased with swarm size, indicating faster convergence as parallel exploration improved. The maximum deflection increased slightly from 0.37mm at three agents to 1.69mm at fifteen agents but remained well below the 2mm serviceability limit, confirming that the deflection constraint effectively maintained global stiffness even under higher loads and structural reach.

Varying the exploration–exploitation decay rate at three agents revealed the expected trade-off between stability and exploration. Slow decay (0.001) produced highly stable but static behavior with no successful spans, whereas faster decay (0.5) enabled greater progress at the expense of slightly higher deflection. The intermediate decay rate (0.1) achieved a 20% success rate and intermediate cantilever length, suggesting a non-linear sensitivity to exploration scheduling. These results support the interpretation that overly conservative exploration limits the agents' ability to extend the structure, while faster exploration accelerates goal discovery within acceptable deformation limits.

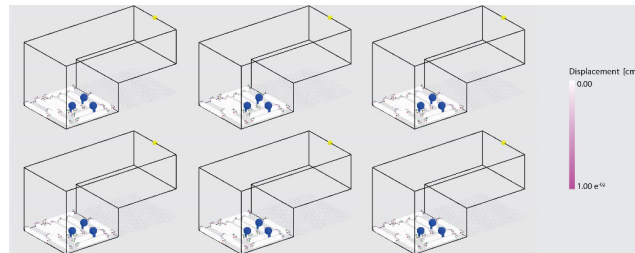


Figure 3. Visualization of the resulting structures from running the ABM considering 3 agents, a decay rate of 0.5 and one rotation option.

Rotation resolution had a similarly constraining effect on performance. With rotation options of 5 and 10, success dropped to zero, accompanied by shorter cantilevers (0.24m and 0.09m) and fewer elements placed, indicating that increasing orientation search space reduced placement efficiency. The best results were obtained with minimal rotation resolution, suggesting that excessive rotational freedom leads to a combinatorial explosion of placement options and more frequent rejection.

Overall, coupling agent decision-making with structural evaluation enabled progressive growth while maintaining serviceability limits. Performance scaled favourably with swarm size, while moderate exploration parameters and constrained rotation options yielded the most stable and efficient emergent assemblies.

5. Conclusion

This paper presents an ABMS approach for CRC that embeds per-placement, deflection-aware FEM into the emergent assembly loop, accepting placements only

when structural limits are respected. An ϵ -greedy policy provides a simple exploration–exploitation balance while a local structural veto steers growth toward mechanically viable configurations without a global blueprint. Rather than verifying structural performance only after completion, structural evaluation is integrated directly into agent decision-making during assembly. The framework remains decentralized, lightweight, and adaptable to changing conditions. Compared to globally planned or centrally optimized workflows, this bottom-up strategy trades geometric optimality for adaptability and parallelism. By decoupling material placement from centralized blueprints, the approach suggests how CRC systems may adapt on the fly to obstacles, material shortages, or evolving design objectives, while bounding structural risk through a minimal global constraint.

The major limitation of this work is that the experimentation is currently conducted in simulation. To further validate the approach, a tighter coupling to the physical world would need to be investigated using one of the proposed methodologies, either with a digital map of the physical world or full decentralization with on-board sensing. As such, material and building tolerances, failure risks, and physical implementation constraints would become major factors to consider in further development.

Beyond physical tests, the ABMS framework also provides a foundation for RL integration. RL could be utilized to enable agents to adapt exploration parameters or learn reusable placement policies. In this way, ABMS defines the interpretable substrate on which adaptive, data-driven CRC control could later be built. Initial experiments show results in combining ABMS and RL (Leder et al., 2024b).

Acknowledgements

This research was supported by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany’s Excellence Strategy – EXC 2120/1 – 390831618.

Attribution Statement

ChatGPT (OpenAI, 2024) was used to improve flow, correct errors in spelling and grammar and improve readability.

References

- Ardiny, H., Witwicki, S., & Mondada, F. (2015). Construction automation with autonomous mobile robots: A review. In *3rd RSI Intl. Conference on Robotics and Mechatronics (ICROM)* (pp. 418–424). IEEE. <https://doi.org/10.1109/ICRoM.2015.7367821>
- Bray, E., & Groß, R. (2021). Distributed self-assembly of cantilevers by force-aware robots. In *Intl. Symposium on Multi-Robot and Multi-Agent Systems (MRS)* (pp. 110–118). IEEE. <https://doi.org/10.1109/MRS50823.2021.9620697>
- Chen, J., & Petersen, K. (2022). Decay-based error correction in collective robotic construction. In *IEEE/RSJ Intl. Conference on Intelligent Robots and Systems (IROS)* (pp. 1–7). IEEE. <https://doi.org/10.1109/IROS47612.2022.9981721>
- Hosmer, T., Mutis, S., Gheorghiu, O., Siedler, P., He, Z., & Erdincir, B. (2024). Autonomous ecologies of construction: Collaborative modular robotic material ecosystems with deep multi-agent reinforcement learning. *Intl. Journal of Architectural Computing*, 22(4), 661–688. <https://doi.org/10.1177/14780771241287827>

- Kayser, M., Cai, L., Bader, C., et al. (2018). FIBERBOTS: Design and digital fabrication of tubular structures using robot swarms. In *Robotic Fabrication in Architecture, Art and Design 2018* (pp. 285–296). https://doi.org/10.1007/978-3-319-92294-2_22
- Leder, S., Kim, H., Oguz, O. S., Kalousdian, N. K., Hartmann, V. N., Menges, A., Toussaint, M., & Sitti, M. (2022). Leveraging building material as part of the in-plane robotic kinematic system for collective construction. *Advanced Science*, 2201524. <https://doi.org/10.1002/adv.202201524>
- Leder, S., Kannenberg, F., Siriwardena, L., Schwinn, T., & Menges, A. (2024a). Approaching the augmentation of heuristic behaviors with reinforcement learning in collective robotic construction. In *SIGraDi 2024 – Biodigital Intelligent Systems: XXVIII Intl. Conference of the Ibero-American Society for Digital Graphics*.
- Leder, S., Siriwardena, L., Kragl, P., & Menges, A. (2024b). Cyber-physical explorations of collective robotic construction using agent-based modelling. In *Robotic Fabrication in Architecture, Art and Design 2024*.
- Leder, S., Kim, H., Sitti, M., & Menges, A. (2024c). Enhanced co-design and evaluation of a collective robotic construction system for the assembly of large-scale in-plane timber structures. *Automation in Construction*, 162, 105390. <https://doi.org/10.1016/j.autcon.2024.105390>
- Melenbrink, N., Werfel, J., & Menges, A. (2020). On-site autonomous construction robots: Towards unsupervised building. *Automation in Construction*, 119, 103312. <https://doi.org/10.1016/j.autcon.2020.103312>
- Melenbrink, N., & Werfel, J. (2017). Local force cues for strength and stability in a distributed robotic construction system. *Swarm Intelligence*, 12(2), 129–153. <https://doi.org/10.1007/s11721-017-0149-2>
- Nguyen, L., Schwinn, T., Groenewolt, A., et al. (2022). ABxM.core: The core libraries of the ABxM framework. <https://doi.org/10.18419/DARUS-2994>
- Petersen, K. H., Napp, N., Stuart-Smith, R., Rus, D., & Kovac, M. (2019). A review of collective robotic construction. *Science Robotics*, 4(28), eaau8479. <https://doi.org/10.1126/scirobotics.aau8479>
- Pietri, S., & Erioli, A. (2017). Fibrous aerial robotics: Study of spiderweb strategies for the design of architectural envelopes using swarms of drones and inflatable formworks. In T. Burry (Ed.), *Proceedings of the 35th Intl. Conference on Education and Research in Computer Aided Architectural Design in Europe (eCAADe)*, Volume 1 (pp. 689–698). eCAADe. <https://doi.org/10.52842/conf.ecaade.2017.1.689>
- Preisinger, C. (2013). Linking structure and parametric geometry. *Architectural Design*, 83(2), 110–113. <https://doi.org/10.1002/ad.1564>
- Stieler, D., Schwinn, T., Leder, S., Maierhofer, M., Kannenberg, F., & Menges, A. (2022). Agent-based modeling and simulation in architecture. *Automation in Construction*, 141, 104426. <https://doi.org/10.1016/j.autcon.2022.104426>
- Swissler, P., & Rubenstein, M. (2022). Reactivebuild: Environment-adaptive self-assembly of amorphous structures. In *Distributed Autonomous Robotic Systems* (pp. 363–375). https://doi.org/10.1007/978-3-030-92790-5_28
- Udaykumar, K., Orozco, L., Krtschil, A., Menges, A., & Knippers, J. (2023). Interactive gradient-based optimization method for column-slab structures. In Y. M. Xie, J. Burry, T. U. Lee, & J. Ma (Eds.), *Integration of Design and Fabrication* (pp. 1574–1584). Intl. Association for Shell and Spatial Structures (IASS).
- Yablonina, M., Kubail Kalousdian, N., & Menges, A. (2020). Designing [with] machines. In *Distributed Proximities: Proceedings of the 40th Annual Conference of the Association for Computer Aided Design in Architecture (ACADIA)* (pp. 330–339).
- Yun, S.-K., Schwager, M., & Rus, D. (2011). Coordinating construction of truss structures using distributed equal-mass partitioning. In *Robotics Research* (pp. 607–623). https://doi.org/10.1007/978-3-642-19457-3_36