Recipe for AI Integration in Architecture

Learning from good practice precedents

Irina Miodragovic Vella [1], Sladjana Markovic[2]
[1]Department of Architecture and Urban Design,
Faculty for the Built Environment, University of Malta
[2] Sustainable and Resilient Architecture Lab,
University of Belgrade - Faculty of Architecture
[1]irina.miodragovic-vella@um.edu.mt [2]marsladjana@gmail.com

Artificial Intelligence (AI) is transforming all disciplines, impacting research, practice, academia, and industry. Architecture is no exception, yet architects have not fully embraced the digital paradigm. AI tools and methods present an opportunity to enhance the positioning of architects within the evolving digital landscape and the increasing complexity and sustainability demands of architecture.
The study proposes a rules-based conceptual framework for an AI-driven, procedure-oriented approach to the design process, informed by historical and contemporary examples of embedded design knowledge as good practice. The recipe for AI integration is derived from an educational methodology illustrated through student projects that combine analogue techniques with digital principles to explore the interplay between algorithmic thinking and AI with a focus on architectural geometry. The paper demonstrates how intuitive investigation, iterative learning, and collaborative processes support the development of computational thinking by embedding design knowledge into procedural systems – introductory for future application. Finally, the paper discusses key guidelines for how AI can be meaningfully integrated into the discipline of architecture.

Keywords*: Artificial Intelligence (AI), Algorithm, Design Knowledge, Positioning of Architect, Architectural Education.*

INTRODUCTION

Artificial Intelligence (AI) is transforming all disciplines, impacting research, practice, academia, and industry. Architecture is no exception. While architects have historically been slow to embrace the digital paradigm fully, AI presents an opportunity, both as a tool and a method, to position architects more effectively within this evolving design landscape (RIBA, 2025) and the future architectural profession as a scientific and societal discipline.

In response to architecture's increasing complexity and pressing sustainability demands, the paper argues that procedure-based architecture and its historical precedents offer a valuable foundation for integrating AI into the design process. It proposes a recipe that draws on traditional architectural approaches to expand the scientific method in service of design knowledge (Ken Friedman, cited in Heylighen et al, 2000).

THEORETICAL BACKGROUND

At the core of procedure-based architecture lies the algorithm, a precise, finite set of rules defining a sequence of operations to solve a problem (Ahlquist and Menges, 2011), where rules are defined as precise, unambiguous geometric statements.

The algorithm enables procedures that embed design knowledge, both the architect's (Schön et al., 1988; Witt, 2010) and collectively established. It makes design knowledge accessible and adaptable, thus susceptible to appropriation, assimilation, and refinement.

From a broader theoretical perspective, procedure-based architecture and algorithmically structured design knowledge are as old as architecture. In his book “The Ten Books on Architecture", Vitruvius offers algorithmic-type instructions on building temple construction (Rowland and Howe, 2001), embedding centuries-old design knowledge in foolproof procedures. Similarly, the key to stone masonry design and construction in the Gothic period required a correct understanding of geometry and the sequences of geometric procedures (Heyman 1995). Later, Antonio Gaudi, Frei Otto, and Hanz Isler developed rigorous, geometry-driven form-finding procedures. These procedures embedded design knowledge about the inherent unity between form and structure that cannot be produced in a single act of artistic creation (Collins, 1963; Burry, 2011; Vrachliotis, at al, 2017, Boller and Schwartz, 2020.).

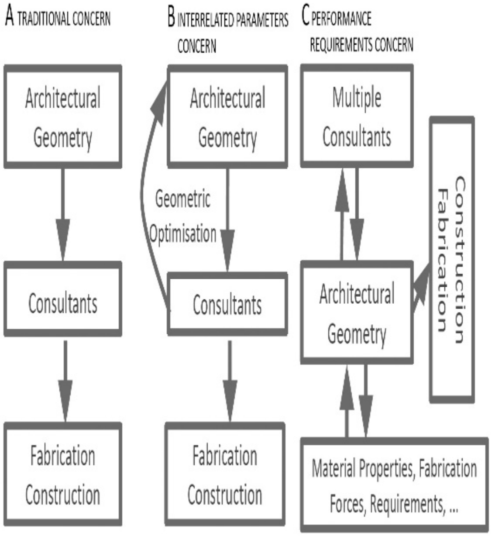
In the contemporary realm, procedure-based architecture was promoted by Christopher Alexander, Cedric Price, and Nicholas Negroponte and MIT's Architecture Machine Group. They argued for the emergence of generative architecture by applying information processes and technologies to architecture. Its procedure-based design process critiqued traditional architectural practices by resisting the development of a specific form or representation as the end goal (Wright Steenson, 2014).

In digital architecture, coding is the procedure that enables architects to embed design knowledge, such as material properties and fabrication constraints, at the outset of the design process (Kolarevic, 2003). The computational machinery interacts in the design process (Winograd, 1997), giving architects dual roles as code creators and users. As a result, they actively contribute to developing and refining design knowledge embedded in these procedures.

Still, the architects have not fully embraced the power of code. Many remain primarily CAD software users, whose software instrumental knowledge and the knowledge that software encapsulates enable or disable the very act of design (Witt, 2010). Additionally, “so much CAD software is pushing a particular semantics of design that is most probably irrelevant except for extremely pragmatic building design and constructed using conventional practice. But lurking underneath the semantics is a very powerful and general geometry toolkit” (Robert Aish, 2005). The challenge remains integrating programming skills into the design process to establish a toolkit capable of creating new design semantics rather than being constrained by predefined ones (Aish 2005). Frequently, the most efficacious solutions combine a variety of digital and AI tools with architect control (Cudzik & Radziszewski. 2018)

The *positioning of the architect* within the design process is constantly evolving. It needs continuous redefinition of activities and characteristics of the architect's role (Markovic, Miodragovic Vella and Hovestadt, 2024).

Figure 1
Relationship approach in design and computational knowledge (adapted from Kloft, 2012)



Research gap

"We ought to be careful about trusting a new technology to create perfect solutions on its own" (Kolatan, cited in Kolarevic, 2003). This caution, voiced during the rise of digital tools three decades ago, resonates with the current enthusiasm surrounding AI in architecture. As with earlier digital innovations, AI tools today generate seductive, visually exuberant "manifest forms" that often overlook the spatial, material, and contextual complexities that define architectural design.

The paper argues that if algorithmic procedures guide generative AI, architects can embed versatile design knowledge into design processes to create meaningful design semantics. Furthermore, in this way, AI can be constructively engaged in addressing future uncertainties to support the development of an architecture rooted in resilience and contingency.

METHODOLOGY

The paper demonstrates this premise through a three-part approach.

First, the paper outlines design workshops based on algorithmic procedures and geometric logic, developed to introduce students to the rule-based design process that embeds design knowledge. The focus shifts from the traditional concern of "how the form is perceived" (Figure 1A) to "how it is generated" through interrelated parameters (Figure 1B). Their methodology follows a five-step iterative procedure that explores multiple solutions rather than a single formal outcome. The procedure was empirically tested, refined, and consistently validated by design workshop outcomes (Miodragovic Vella and Markovic, 2024; Miodragovic Vella, 2023).

Second, the paper identifies success criteria for evaluating workshop outcomes, which, together with key insights, form the basis for a recipe for AI integration.

Third, this recipe informs a speculative framework for AI use in architectural design.

Finally, the paper proposes a design workshop as a testing sample to validate the proposed AI framework.

Iterative Five-Step Procedure

The five iterative steps are formulated from historical and contemporary examples of embedded design knowledge in design procedures.

1. **Parameters investigation:** Identifying and analysing relevant parameters and their interdependencies to define the design problem. It encourages a shift from form-based design to systemic thinking, forming the foundation for algorithmic structuring.
2. **Rule definition:** Investigation to formulate precise geometric logic as rules addressing the specific design standards associated with each parameter. The rules embody contextual logic and design knowledge that both human and machine systems can execute.
3. **Geometric method statement:** Translation of the geometric logic as an algorithm that

expresses the design process and generates a range of solutions. It solidifies computational thinking by externalising abstract knowledge into structured logic.

4. **Catalogue creation:** Executing the algorithm and systematically recording its variations to be evaluated, selected, and refined.
5. **Architectural application:** Selection and application of variations to evaluate how well procedural knowledge adapts to real-world complexity, performance constraints, and material conditions.

Thinking, Tools, and Working Methods

The methodology uses computational thinking to guide design development through rules-based geometric logic (Steps 1 & 2). The geometric method statement (Step 3) is a tool for introducing this mindset to prompt articulation of the design intent using simple, precise geometric principles. The approach improves clarity, minimises ambiguity and errors, and deepens understanding of geometric complexity in architectural design (Peters, 2008).

The method statement is formulated as an algorithm expressed diagrammatically. Its multiple executions (Step 4) allow evaluation of outputs' performance and adaptability and a clear and structured assessment of the design's procedural logic (Step 5). The result is a positive feedback loop in which abstract design knowledge translates into operative geometric tools and vice versa.

The construction of the geometric method statement combines analogue and digital techniques. Initially, they externalise design knowledge related to parameters and interdependencies (Step 1), then test, iterate, and refine the geometric logic (Step 2). Analogue techniques include model-making, sketching and pseudocode. Digital approaches comprise Rhino 3D modelling and Grasshopper parametric modelling. In this way, CAD software is a visualisation tool and a medium for constructive form generation and evaluation. The emphasis on specific techniques varies according to the focus and objectives of the design process.

Similarly, the duration and investigative scope of the design process determine the choice between individual and group work.

The evaluation revealed the degree to which the success criteria were met was tied to the design process's tools, techniques, methods, and duration. These were consolidated into four key aspects for AI integration: investigation, data, work, and duration.

DESIGN WORKSHOP RESULTS

The methodology structured two design workshops that introduced students to procedure-based architecture. The five iterative steps allowed students to engage gained design knowledge to inform their design decisions, further explorations, and evaluation processes.

The task of both design workshops was to develop a shading façade screen for the University of Malta Library building.

The first workshop spanned a semester (14 weeks) and prioritised individual work and digital tools, primarily Grasshopper for parametric modelling and Ladybug for solar analysis.

The second workshop was shorter (7 weeks) and emphasised analogue investigation techniques and group work.

Design Process

The aim is to establish a design process derived from, and responsive to, multiple contextual parameters. The scope was to define a procedure that embedded and activated design knowledge in response to the screen's key performance requirements (Figure 1C): sun exposure, natural light requirements of the library programme, load paths, materiality, and aesthetic expression.

Following Step 1, which focused on parameter investigation, students applied regular tessellation rules to establish a procedural design logic that guided the formulation of geometric rules defining the screen's design intent. The rules included parametric variations of tile geometry, enabling the screen to mediate between sun exposure and programmatic requirements.

In Step 2, the geometric rules incorporated additional information from other parameters, including material properties, fabrication constraints, and the assembly sequence. The students tested the versatility and robustness of their design process through cataloguing, contextual visualisation, and the design of related architectural elements (Figure 2).

Design Workshop Outcomes

The five selected student projects from the second workshop (Figure 3) demonstrate how the methodology fostered design procedures that enabled design knowledge to guide project development from concept to architectural scale. Each project fulfilled the workshop task through a distinct rule-building strategy, ranging from simple to complex and employed a different mode of design representation (Lawson, 2005). The projects developed procedures driven by aesthetics, material behaviour, and structural logic by prioritising specific design parameters.

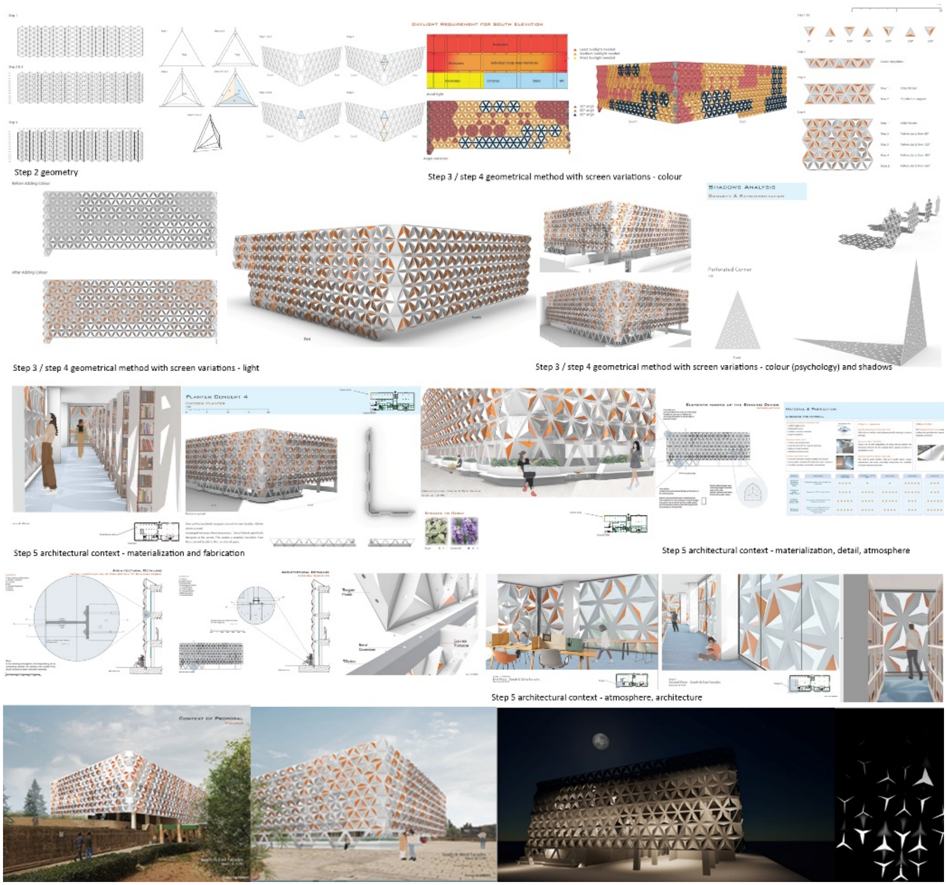


Figure 2
Computational design process tests: from cataloguing and contextual visualisation to design of architectural elements,
Group 2: Pace, C., Hili, E., Sammut, K., Cassar, M.

Group 1 (Figure 3a) and Group 4 (Figure 3d) prioritised aesthetics in Steps 2 and 3 rather than optimising shading performance. They revised their geometric logic to align design intent with fabrication and assembly requirements, replacing tessellation with a continuous strip-based geometry. Group 4 also abandoned parametric variability in favour of a layering strategy to control screen permeability.

Group 2 (Figure 3b) applied insights from Step 1 to optimise material use and ease of fabrication and assembly. The outcome was a performative yet restrained tessellation. The colour applied in a seemingly random but procedurally controlled way introduced a sense of playfulness.

Group 3 (Figure 3c) investigated structural folding to derive geometric rules for a tessellation of volumetric tiles with strong three-dimensional expression. Embedded design knowledge guided the development of a performative, self-supporting geometry, enabling efficient assembly and resulting in a distinctive aesthetic.

Together, the design workshop outcomes demonstrate the methodological impact and versatility of the rule-based approach in supporting diverse design strategies. Serving as both pedagogical experiments and empirical case studies, the outcomes and insights informed the development of a "recipe" for AI integration and validated its procedural and cognitive dimensions

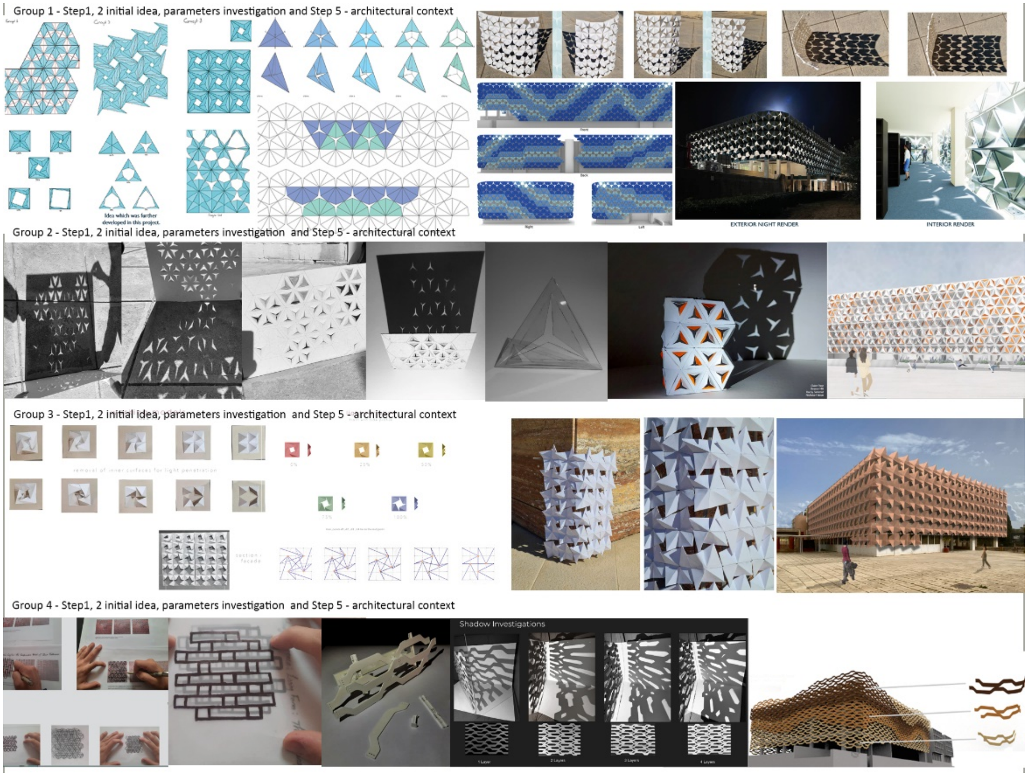


Figure 3 Design procedures: design knowledge guiding project development from concept to architectural scale. Group 1: Vassallo, J., Govus, J., Spiteri, M., Tonna, M. Group 2: Pace, C., Hili, E., Sammut, K., Cassar, M. Group 3: Muscat, V, Riolo, N., Agius, M., Agius, L., Group 4: Cremona, C., Azzopardi, J., Garcia, M., Mohammed, N.

RECIPE FOR AI INTEGRATION

The comparative evaluation of the design workshop outcomes used five criteria: (1) innovation and creativity; (2) clarity and robustness of the procedural logic; (3) the ability to synthesise data as design information; (4) depth of embedded design knowledge; and (5) response to performance requirements. Success in meeting the criteria correlates with how the design process approached investigation, data, work, and duration. The four key insights underpin the proposed recipe for AI integration.

Investigations: Digital vs Hands-on

Learning software and programming to support design investigations requires motivation and focus, which are not always available. When only limited coding and software skills are acquired, investigations tend to rely on a narrow set of commands, restricting design exploration and the development of computational thinking. The resulting design process offers reduced creativity and lower engagement.

In contrast, hands-on and familiar investigation techniques offer an intuitive and accessible approach. They lead to successful procedure generation and embedding of versatile design knowledge into design processes. The result is meaningful design semantics and the fostering of computational thinking as a cognitive process for creating artefacts (Mortati, 2022).

Integrated into intuitive investigations, AI allows quick visualisation of speculative environmental performance, geometric variations, and material tests, enhancing the validity and adaptability of the design logic without displacing the architect's authorship.

Data: Precision vs Design Information

Speculative or imperfect data can be more valuable than precise data. The key lies in understanding and translating data into design knowledge that informs procedures. Data from external sources often lacks transparency about how it was measured or collected, making it prone to misinterpretation despite its accuracy. In contrast, actively acquiring and interpreting data offers greater control and enables deeper, context-specific understanding.

Integrated in this approach to data, AI fosters data literacy and procedural translation that assist in structuring and interpreting design-relevant information.

Work: Individual vs Group

While individual work supports stronger authorship and design coherence, it often leads to narrower, linear investigations focused on a single parameter. In contrast, group work promotes peer-to-peer knowledge exchange, allowing individuals to take on responsibilities aligned with their skills and interests. It fosters collective ownership while maintaining individual creative agency.

Collaboration allows parallel investigation of multiple parameters and their interdependencies and diverse perspectives to inform rule development. It leads to precise geometric method statements and versatile design procedures.

Integrated into group work, AI acts as a collaborative platform, responding to varied inputs and enhancing personal interpretation within a shared procedural logic essential for fast feedback loops between human intuition and AI-generated variation.

Duration: Short vs Long

Procedural methods are effective when applied in iterative cycles within shorter timeframes. Time limits encourage faster decisions, sustained focus, and steady progress through each procedural step. The higher iteration rate accelerates learning, particularly in collaborative contexts. Short, focused rule-building sessions enhance procedural clarity and adaptability. In contrast, longer timelines often result in procrastination, over-analysis, and slower engagement.

Integrated into short, focused sessions, AI tools assist in rapidly producing and evaluating design variations based on user-defined criteria. They support high-frequency iteration, which is essential for training the designer's intuition and potential AI feedback systems.

The four key aspects of the recipe investigation, data, work, and duration, directly inform the framework for AI integration, which builds on the procedure-based methodology by addressing each identified aspect.

FRAMEWORK FOR AI INTEGRATION

The framework for an AI-driven, procedure-oriented architectural design process explores the interplay between algorithmic thinking and AI.

Intuitive Investigations

The recipe emphasises hands-on, exploratory, open-ended, and non-linear investigations, analogue or digital, as the primary means of acquiring design knowledge. AI, as a tool and a method, can support these qualities if engaged through accessible and familiar modes of interaction. It allows designers to retain control of their design intent within procedural frameworks. Rather than dominating the design flow, AI is a supportive tool, helping embed design knowledge into geometry-based algorithms. Design decisions remain grounded in design knowledge and preserve the personal touch and human qualities essential to architectural thinking (Markovic, Miodragovic Vella, Hovestadt, 2024).

Imprecise Data

The recipe positions AI as a tool and a method for acquiring data in accessible, actionable ways. It enables data interpretation, reframing, and translation into design knowledge that informs procedural development. Rather than offering ready-made answers, AI enhances exploratory, human-led design thinking. The design outputs serve as data for training AI in subsequent iterations, allowing it to remain responsive to the evolving design intent and underlying logic. Though initially limited, the approach gains momentum as iterations build.

AI engages with each procedural step: investigating multiple parameters and their interdependencies, formulating geometric method statements, and supporting its cataloguing and contextual applications. AI becomes an extension of the designer's creative capacity, responding to contextual complexity, contingency, and the unknown.

Personal Collaboration

The recipe proposes AI as a collaborative platform that enhances personalised engagement within collective design processes. Unlike rigid methods and identical prompts of traditional software, AI adapts to diverse verbal, visual, spatial, or procedural inputs, accommodating different thinking styles. The flexibility preserves individuality within group work.

Rather than flattening differences, AI amplifies personal interpretation within a shared procedural logic. It supports formulating, testing, and refining geometric method statements and design procedures shaped by diverse, personalised contributions.

Iterative Learning

The recipe engages AI in gaining design knowledge through investigation and embedding it into dynamic, responsive design procedures. It encourages high iteration rates, as frequent steps boost engagement, accelerate decision-making, and strengthen computational thinking, enhancing designer and AI learning. Repetition deepens procedural insight, while AI adapts through exposure to varied design inputs. The iterative rhythm validates the geometric logic and design procedures it informs. In this context, AI supports the rapid testing of geometric rules and evaluation of catalogue outputs, reinforcing iteration as a driver of procedural clarity and embedded design knowledge.

Design Thinking from Good Practice

Knowledge of historical precedents and contemporary advances should remain in active dialogue, allowing for the continual re-examination and refinement of design processes. As design knowledge evolves with technology, it emerges across multiple architectural layers. Today, being an architect increasingly involves embodying the Vitruvian ideal: a fully integrated artist and engineer capable of multidisciplinary work (Markovic, 2024). The role requires a synthesis of theory (to remain grounded in context), art (to embrace technological innovation), and engineering (to collaborate across disciplines, including with nature), empowering architects to observe, develop, and advance complex design knowledge and skills.

Reflection Criteria' Sample

Workshop outcomes were also evaluated against five reflection criteria derived from the AI integration framework: (1) translating contextual understanding into structured procedural logic; (2) reframing data as design knowledge; (3) supporting both collaborative exploration and individual authorship; (4) fostering iterative learning and adaptability; and (5) producing catalogues for architectural decision-making.

The charrette-style workshop pilot tests a five-step AI-integrated methodology within a one-day, three-task format: hands-on physical modeling, geometric logic and catalogue creation, and material-context exploration. Each task ends with a pin-up for process evaluation. AI tools support rule formulation, variation generation, and visualisation. Outcomes are assessed using five reflection criteria: translating context into logic, reframing data as design knowledge, balancing collaboration and authorship, fostering iterative learning, and producing decision-informing catalogues. The format emphasizes rapid prototyping, adaptability, and immediate feedback. This pilot evaluates AI's potential in supporting creative workflows and informs the design of a longer seven-week studio format.

CONCLUSION

AI does not need to replace architects but can enhance architectural thinking when aligned with design knowledge and algorithmic procedures. This study introduces a methodology rooted in computational logic and workshop insights, offering a framework for integrating AI through intuition, data engagement, collaboration, and procedural iteration. The approach fosters exploratory thinking and contextual reasoning by embedding design knowledge into rule-based steps. Empirical evidence underscores the value of procedural design and cognitive processes alongside technical systems. Future work will expand the framework to evolving AI applications, where AI acts as a collaborative

REFERENCES

Ahlquist, S. and Menges, A. (2011) *Computational Design Thinking: Computation Design Thinking*. Chichester: John Wiley & Sons.

Aish, R. (2005) 'From intuition to precision', *AA Files*, 52, pp. 62–63.

Boller, G. and Schwartz, J. (2020) ‚Modelling the form: Heinz Isler, Frei Otto and their approaches to form-finding', *Seventh Conference of the Construction History Society*, Cambridge, pp. 565–776.

Burry, M. (2011) 'Geometry Working Beyond Effect', *Architectural Design*, 81(4), pp. 80–89. Available at: https://doi.org/10.1002/ad.1272

Collins, G.R. (1963) *Antonio Gaudi: Structure and form*. The MIT Press, 8, pp. 63–90.

Cudzik, J. and Radziszewski, K. (2018) 'Artificial Intelligence Aided Architectural Design', *eCAADe 36*, Vol. 1, pp. 77–84.

Heylighen, A., Neuckermans, H. and Hanrot, S. (2000) 'Design(ing) knowledge in

architecture', *Recherche et Architecture / Research and Architecture*, Paris, pp. 231–241.
Heyman, J. (1995) *The Stone Skeleton: Structural Engineering of Masonry Architecture*. Cambridge: Cambridge University Press.
Kloft, H. (2012).'Engineering Freeform Architecture', in Coarser, R. (Ed.), *Fabricating Architecture*, pp. 110–128. Princeton, NY: Princeton Architectural Press.
Kolarevic, B. (2003) *Architecture in the Digital Age: Design and Manufacturing*. New York and London: Taylor & Francis.
Lawson, B. (2005) *How Designers Think*. New York: Routledge.
Markovic, S. (2024) *Logbook of Challenges in Contemporary Positioning of Architect – the Principle of "Digital Chain" and Beyond...* (in Serbian). Belgrade: University of Belgrade, Faculty of Architecture.
Markovic, S., Miodragovic Vella, I. and Hovestadt, L. (2024) 'A conceptual framework for positioning the architect within continual digital approach to designing architecture: the 'digital chain' case', *Facta Universitatis – Series: Architecture and Civil Engineering*. Available at: https://doi.org/10.2298/FUACE240713002
Miodragovic Vella, I. and Markovic, S. (2024) 'Topological Interlocking Assembly: Introduction to Computational Architecture', *Applied Sciences*, 14(15), 6409. Available at: https://www.mdpi.com/2076-3417/14/15/6409
Miodragovic Vella, I. (2023) 'Digital design in 'peripheral' contexts', *eCAADe 41*, Vol. 1, pp. 61–68.
Mortati, M. (2022) 'New Design Knowledge and the Fifth Order of Design', *Design Issues*, 38(4), pp. 21–34. Available at: https://doi.org/10.1162/desi_a_00695
Peters, B. (2008) 'The Copenhagen Elephant House: a case study of digital design processes', in *Biological Processes and Computation: Proceedings of the ACADIA 28th Conference*, pp. 134–141.
RIBA (2024) *Artificial Intelligence: How are architects using AI right now and what are they using it for?* Available at: https://www.architecture.com/knowledge-and-resources/knowledge-landing-page/artificial-intelligence-in-architecture (Accessed: 30 January 2025).
Rowland, I.D. and Howe, T.N. (Eds.) (2001) *Vitruvius: Ten Books on Architecture*. Cambridge: Cambridge University Press.
Shoen, E., Smith, R.G. and Buchanan, B.G. (1988) 'Design of Knowledge-Based Systems with a Knowledge-Based Assistant', *IEEE Transactions on Software Engineering*, 14(12), pp. 1771–1791.
Vrachliotis, G., Kleinmanns, J., Kunz, M., Kunz, P. (Eds.) and Otto, F. (2017) *Frei Otto – Thinking by Modeling*. Leipzig: Spector Books.
Winograd, T. (1997). 'From Computing Machinery to Interaction Design', in Denning, P. and Metcalfe, R. (Eds.), Beyond Calculation. Springer-Verlag, pp. 149–162.
Witt, A. (2010) 'A Machine Epistemology in Architecture' (translated by Annette Wiethuchter), *Candide: Journal for Architectural Knowledge*, 3(12), pp. 37–88.
Wright Steenson, M. (2014) *Architectures of Information: Christopher Alexander, Cedric Price, and Nicholas Negroponte and MIT's Architecture Machine Group*. Princeton, NJ: Princeton University.