

Rapid Prototyping for Kinetic Architecture

Peter Ferschin, Monika Di Angelo, Gerhard Brunner

Abstract— Architecture was perceived for many centuries as the design of static buildings that will last for extended periods of time, with static changes or additions at most. Traditional design tools reflect this approach. Kinetic architecture, however, defines a kind of architecture that incorporates dynamic properties which accommodate for environmental or utilization changes. The integration of mechanics, electronics, and computer technology into contemporary architectural design, requires a change of thinking and designing, as well as suitable design tools. To communicate design ideas, the architectural design process utilizes effective strategies such as sketching and model building. A well-established design technique in diverse disciplines is rapid prototyping. Our goal is to have an integrated environment for designing kinetic architecture, including its mechanical parts, electronic units, and software modules with a rapid prototyping approach. This paper discusses conceptual ideas for rapid prototyping design environments for kinetic architecture. An exemplary environment is introduced where a small design task has been realized to provide feedback for the concept.

Keywords—rapid prototyping; kinetic architecture; design process.

I. INTRODUCTION

Kinetic architecture is more and more converging towards mechatronic design, as demonstrated e.g. in the Reichstag Dome in Berlin [1, 2]. Fig. 1 shows how natural light is tunneled into the interior of the Reichstag through the steel and glass construction, while a moving shade controls the amount of light.

There is a highly visible trend for buildings to get wired (i.e. to use sensors, actuators, and bus systems) on all levels from home automation, over smart buildings to kinetic architecture. “The kernel of interactive architecture is the integration of mechanics, electronics, and computers in a robotic building module that can sense and transform itself in response to change.” [3]

A common aspect of robotic, kinetic, interactive or responsive architecture and smart buildings is to adapt intelligently to environmental changes like day light, wind, and temperature. The terms kinetic, interactive, robotic, and responsive architecture as well as smart buildings widely overlap and are used more or less synonymously or with a slightly different emphasis. In this paper we use kinetic architecture to underline its physical changes.

P. Ferschin is with the Center for Geometry and Computational Design, University of Technology, Vienna, Austria (phone: +43(1)58801-27210; fax: +43(1)58801-27299; e-mail: peter.ferschin@tuwien.ac.at).

M. Di Angelo and G. Brunner are with the Department for Computer-Aided Automation, University of Technology, Vienna, Austria (e-mail: monika.diangelo@tuwien.ac.at).



Figure 1. Reichstag Dome in Berlin by Foster + Partners, steel and glass structures by Waagner-Biro, www.waagner-biro.com

Another prominent example is depicted in fig. 2, the Burke Brise Soleil, “a moveable sunscreen with a 217-foot wingspan that unfolds and folds twice daily” and “its two ultrasonic wind sensors automatically close the wings if the wind speed reaches 23 mph or greater.” [4]

Designing kinetic architecture is still a challenge and not well supported by available computer-aided architectural design tools since they lack design support for moving elements, electronic parts and software components. They require complex data exchange with other design and simulation tools, which leads to an extensive planning effort. A better integration of architectural and mechatronic design environments would be desirable.

Rapid prototyping is a well-established design technique to evaluate the quality of a design and its feasibility early on. “Foster + Partners have been one of the first architecture practices to fully integrate rapid prototyping within its design process. ... With the introduction of in-house rapid prototyping machines, the architectural design process within the office has changed.” [5] So far, the incorporation of rapid



Figure 2. Burke Brise Soleil of the Quadracci Pavilion, addition to the Milwaukee Art Museum by Santiago Calatrava, calatrava.com

prototyping into the architectural design process seems to be limited to 3D printing, though [5, 6].

The increase in accessibility and affordability of electronic prototyping environments (e.g. Arduino) further encouraged the development of kinetic architecture, as sketching and prototyping of the involved electronics became ever more feasible. Architectural design tools pick up the trend by providing a link to electronic environments (e.g. Grasshopper extensions for Arduino). However, they are still lacking a full support to “explore the dynamics of architectural space by re-thinking architecture beyond conventional static and single-function spatial design. Adaptive response to change should be considered in the preliminary design process prior to the stage of construction and use.” [3]

This early stage of design requires interdisciplinary thinking as well as new tools for rapid prototyping in order to fully support kinetic architecture. Our goal is to have an integrated environment for designing kinetic architecture, including its mechanical parts, electronic control units, and software modules for the intended behavior of the designed kinetic building with a rapid prototyping approach.

II. REQUIREMENTS FOR A DESIGN ENVIRONMENT

A design process for kinetic architecture has to take manifold and complex requirements into account. Kinetic architecture not only deals with movable parts added to static architecture, it also involves a shift of thinking in the design process towards viewing a building in a process of change. This includes a strategy for the kinetic movements in order to orchestrate the desired physical adaptability of buildings. The motivation for adaptability is currently dominated by energy efficiency, living comfort, or higher percentage of utilization [7-12].

Rapid prototyping is an elaborate manufacturing technology [13], using physical as well as virtual prototypes with manifold application areas. They range from virtual environments in mechatronics [14-17] and courses on embedded systems [18], over motion prototyping [19] to physical prototypes in interface design [20] and virtual-physical prototyping in the design of media facades [21].

C. Hoberman is renowned for his pioneering work in transformable design [22]. Ideas on interactive architecture as an interdisciplinary design process are summarized by M. Fox in [23].

A suitable rapid prototyping environment for kinetic architecture has to adequately support the design process by providing for modelling, visualization, simulation, control, and evaluation of the virtual and physical prototypes throughout the entire design process. This implies above all, an iterative design process with repeated evaluation of suitability and aesthetic qualities.

For kinetic architecture being complex by nature, it is indispensable to create virtual as well as physical prototypes that explore the functional aspects of the design goal. As opposed to traditional media for architectural expression like paper or physical models, an (additional) virtual model provides viable means to explore the dynamic and interactive qualities of an object in a fast manner. In order to evaluate the feasibility of a kinetic design, a physical functional prototype

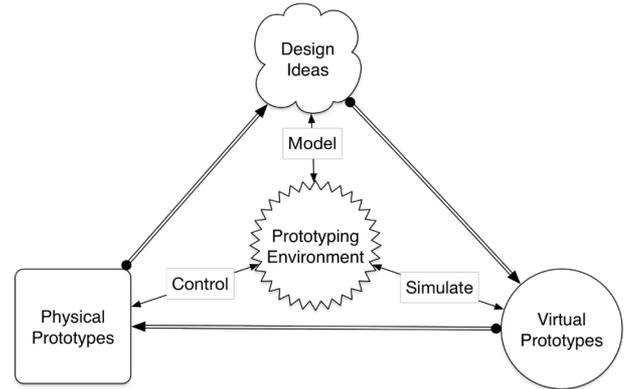


Figure 3. Design process for kinetic architecture

is required as well, that comprises mechanical, electronic and software components.

Therefore, we propose a design process for kinetic architecture that consists of three major stages, as depicted in fig. 3: The first stage comprises the sketching of design ideas and goals. In the second stage, a virtual prototype is built, that allows the simulation of functional aspects. In stage three, the feasibility of the design can be evaluated with a functional physical prototype. All stages provide feedback to iteratively improve the design. The ideal prototyping environment for kinetic architecture provides support for all three stages, and combines them into a unified system. To facilitate fast iteration cycles, the interchange between the stages should be as seamless as possible.

Furthermore, rapid prototyping for kinetic architecture is only feasible when it is fast and low-cost on every level. Additionally, it should allow for the re-use of previously designed components of similar functional solutions. It should provide pre-built components that are easily extended with new design ideas.

Major tasks of the prototyping environment are to model and visualize the design idea, and to simulate and control its functional aspects in the virtual and physical prototypes. Ideally, it facilitates the evaluation of the design according to the specified design goals.

The design of kinetic architecture can be structured into a layered approach as depicted in fig. 4. We differentiate the physical layer into the static and mechanical parts. The mechanical parts comprise both manually and electronically operated objects. The integration of electronics and software has to be considered as well.

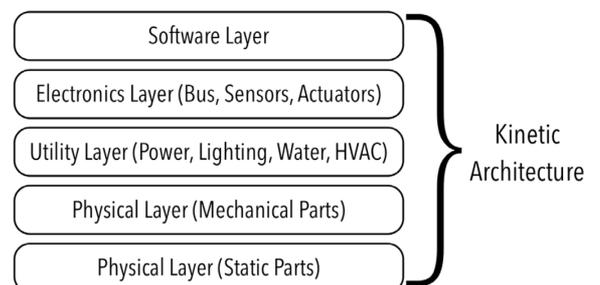


Figure 4. Kinetic architecture layers

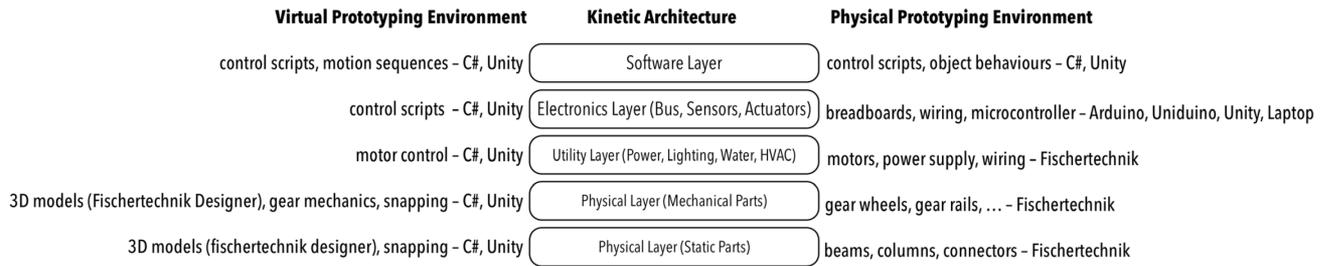


Figure 5. Kinetic architecture layers with exemplary prototyping environments

III. PROTOTYPING FRAMEWORK

For designing kinetic architecture, the avant-garde architectural design community uses different approaches with a variety of prototyping frameworks. For virtual models a combination of existing tools is used, such as Rhino + Grasshopper, Processing, Blender + Python scripting, Vrttools or similar environments. For electronic prototyping, Arduino is the most popular environment. Some approaches use LabView with custom electronic components. Due to the lack of unified environments, there are several custom solutions for proprietary use only. Physical prototypes are created with traditional model building efforts, and increasingly by using 3D printing.

In the search for a unified design and rapid prototyping environment we realized our first approach as depicted in fig. 5. The real-time game engine Unity was selected as a suitable development platform for the virtual design and simulation prototype and the control of the physical prototype for the following reasons:

- well suited for design-oriented and technology-oriented users, because of tight integration of design software (images, audio, 3D models, GUI) and a software development environment
- flat learning curve
- extensible with design methods, software components for simulation and additional editing functions
- excellent interfacing possibilities to various hardware devices, like Arduino

For physical prototyping we resorted to popular building systems. After experimenting with Lego Mindstorms, we tried Fischertechnik because of insufficient precision in the mechanical parts. For controlling the physical prototype Arduino was chosen because of its popularity in the physical computing domain. Uniduino, a Unity extension for Arduino,

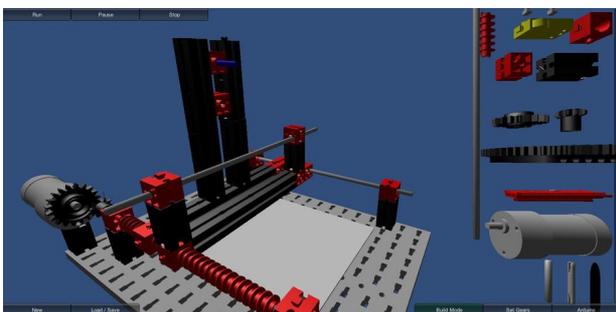


Figure 6. Fischertechnik parts in the virtual prototype

was used to interface Unity with Arduino. The Fischertechnik Designer provided 3D models of Fischertechnik parts, which were converted to the Unity environment.

A. Modelling

For the physical prototype the static parts were modelled in Fischertechnik bricks and beams etc., the mechanical parts with Fischertechnik cog wheels, gear rails, and so on. The utility layer, too, was modelled with Fischertechnik and wired to the next layer. The electronics layer was modelled with Arduino and controlled by Unity which represents the software layer.

Fig. 6 depicts the virtual design environment showing available Fischertechnik parts on the right side and the actual design on the left side. Like physical Fischertechnik parts, they can be plugged together using the provided connectors. Software design constraints ensure that parts can only be fitted together in a meaningful way.

When a part is placed in a close range to another part, they are snapped together, i.e. the part being placed is automatically moved to the nearest possible connection point. They user can accept this proposed connection.

B. Simulation

If a motor of a mechanical construction is activated in the virtual environment, its behavior is simulated. If the physical prototype is built in a matching way, its behavior can be directly controlled from the virtual environment.

This first approach towards a unified prototyping framework uses a real-time environment for the virtual prototype. We strongly believe that this is an indispensable key concept for designing kinetic structures. Real-time simulation is not only necessary for controlling the physical prototype, it is essential for realistic feedback in the design process of the kinetic elements.

Another interesting aspect lies in mechanisms for the transfer of motion. Although the chosen implementation platform contains a sophisticated physics engine for game

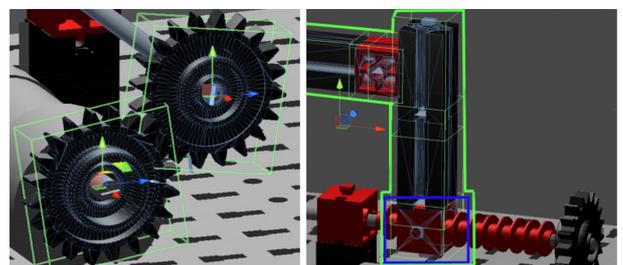


Figure 7. Left: cog wheel interaction, right: object hierarchy

development, it was viable to implement the virtual physics in a purpose tailored way as a reduced model of mechanical motions. In the sense of rapid prototyping, a fast feedback of the functional aspects of the prototype has a higher priority than accuracy in mechanics modelling, similar to game engine physics in general.

The reduced mechanics model uses the collision detection provided by the development environment Unity for the identification of a physical interaction. The actual movements, though, are tailored to the involved parts, e.g. motion transfer between cog wheels, for simplicity and increase of calculation speed. This has been implemented by a twofold geometrical representation of the physical objects. A detailed geometric model was used for rendering and a simplified geometric model (a bounding box) for transferring the motion, as depicted on the left side of fig. 7. The transfer of motion was accomplished by implementing parametric gear mechanics, with parameter like rotation speed or number of cogs in a wheel.

The object hierarchy is depicted on the right side of fig. 7, whereby static objects, which are connected to a moving part, are grouped together as sub-objects of the moving part as indicated by the green border. This ensures in a simple way that the motion of a moving part is transferred correctly to its connected static parts.

C. Control

Furthermore, focus was put on the transfer of control from the virtual to the physical prototype. In the virtual environment, the motor movements can be defined as a sequence diagram, where the parameters are the movement direction and the duration of a continuous motor movement.

In a setup phase, the virtual motors have to be connected to their physical counterparts. This is performed by connecting virtual pins of the virtual motor to physical pins on the Arduino board, which is the interface to the physical motors, and which has to be connected as well.

Then the physical prototype can be controlled from the virtual environment by transferring the movement sequences to the physical motors.

IV. EXEMPLARY DESIGN

A simple test case was used to evaluate the proposed design environment, the design of a 2D plotter. This choice put a focus on the first implementation of the design environment, instead of being distracted by a complex architectural design task.

After sketching the plotter (cf. fig. 8) in a few design variants, a virtual version of the plotter was built. Each functional component was tested in the virtual environment after building (e.g. testing the transmission of the gear rail via motor rotation). Then a motion sequence for plotting was defined. The plotter was tested iteratively and redesigned if necessary. Then the physical plotter had to be built and connected.

Two iterations were realized as virtual and physical prototypes. The first iteration is depicted on the left side of fig. 9, the second iteration is shown on the right side. For evaluation purposes, the design times have been recorded:

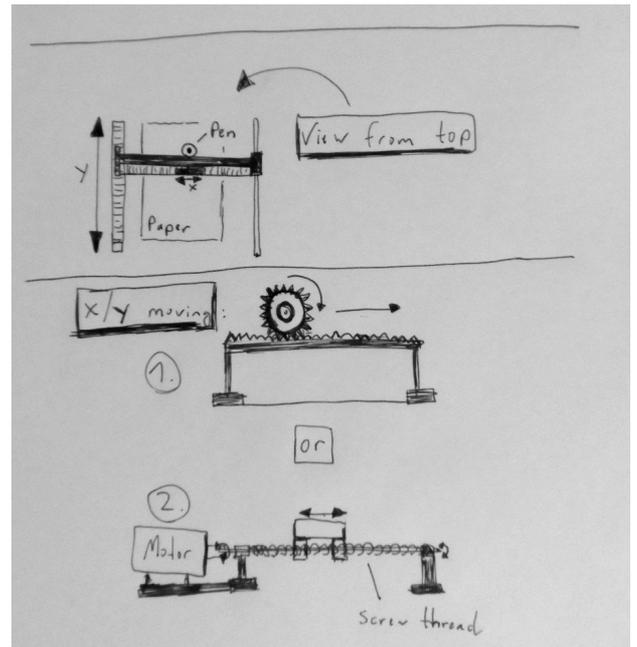


Figure 8. Plotter sketch

In the first iteration, the time required for the virtual prototype of the 2D plotter was 30 minutes. To build the physical 2D plotter with Fischertechnik parts, 60 minutes of time were invested. Constructing the plotter in a stable and solid manner took more time than expected. This was not obvious from the virtual environment.

In the second iteration, it took 15 minutes to remove pieces from the virtual prototype and rebuild the top part of the plotter. Rebuilding the physical 2D plotter took also 15 minutes. The second iteration addressed stability and precision issues of the first construction, by changing the position of the motor and the size of the cog wheel.

The set-up of the test case is depicted in fig. 10 with the virtual prototype on the laptop, the physical prototype as a Fischertechnik construction, and Arduino as the connecting interface.

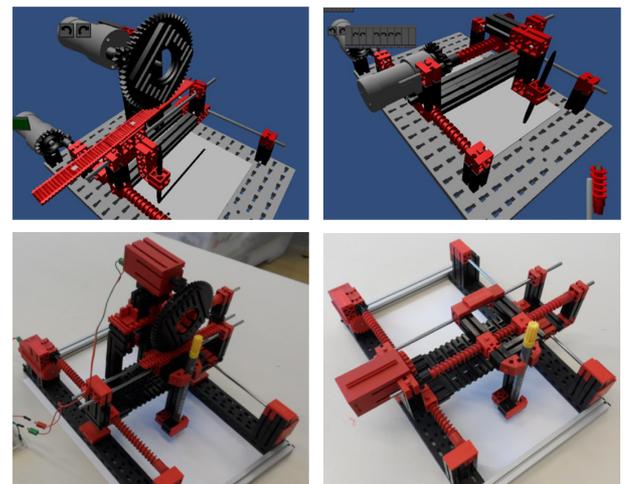


Figure 9. Plotter Prototypes, left: iteration 1, right: iteration 2
upper row: virtual prototypes, lower row: physical prototypes

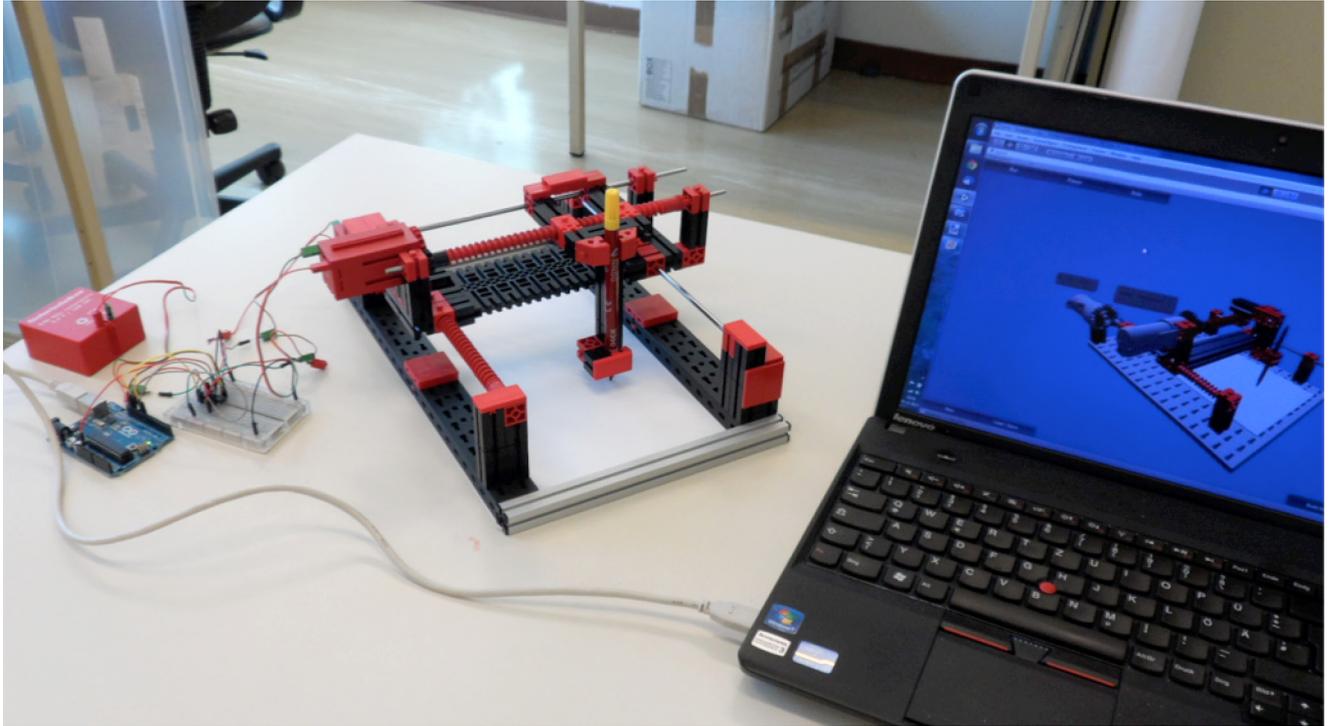


Figure 10. Combined virtual and physical prototyping environment

V. RESULTS AND DISCUSSION

With the presented first approach towards a rapid prototyping design environment, some steps of the design process for kinetic architecture could be implemented as a proof of concept.

In the very early stage of design, the sketching of the design ideas was still performed in the traditional way of pen and paper.

The stage of creating a virtual prototype was achieved by providing a toolbox system for predefined objects, which was extensible by concept. To aid the design process, several constraints for positioning the objects from the toolbox were implemented. Snapping of objects to possible connecting elements occurred wherever it was reasonable.

The stage of physical prototyping was realized by using Fischertechnik sets that included motors. For the integration of the virtual and physical prototype, a standard Arduino kit proved suitable.

We could show that the integration of virtual and physical prototyping provides both valuable feedback in the design process and fast iteration cycles.

As we argue for rapid prototyping, a simplification of functional aspects is necessary to provide fast feedback in the design process. The virtual prototype represents a higher abstraction, and needs verification and evaluation by the physical prototype.

The specification of the behavior of the kinetic construction was implemented as a graphical sequence of motor movements. The simulation of a specified behavior could be performed in the virtual prototype including all dependent mechanical movements (e.g. connected parts and

parts to which forces have been transferred). The same motion sequence could be directly transferred to the physical prototype, thus controlling the physical prototype from the design environment.

Parts of the design could not suitably be realized with the provided components, e.g. the pen holder for the plotter, because the available Fischertechnik sets did not comprise the necessary parts. Instead, this required a custom part both in the virtual and physical prototype. In this first approach, the integration of custom parts was generally possible, but lacked a comfortable support for the user.

Several aspects have not been accounted for in this first approach and are still open issues:

The transfer of behavior from the virtual to the physical prototype was possible, but several fundamental problems became visible: The virtual construction was faster and simpler, since stability was not an obvious issue in the virtual world. Generally, the virtual and physical prototypes easily diverge because of mechanical problems and simulation details. Particularly the Fischertechnik parts proved to lack the desired precision in some aspects.

The synchronization between physical and virtual prototype with respect to movements needs major improvement, e.g. via sensory feedback. This would be possible with our approach, but has not been implemented yet.

New parts in the virtual environment should be available in the physical environment as well. This could be achieved by 3D printing or CNC milling for most parts. Parametric modelling tools with integration into digital fabrication systems could be used for custom parts such as the pen holder, which was already part of our concept.

Parts that are added in the physical environment have to be appropriately represented in the virtual environment. This could be achieved by 3D scanning or 3D modelling. Additionally, these models have to be prepared for a real-time environment with respect to geometry and functional aspects.

Similar integration methods have to be devised for electronic parts, such as additional sensors and actuators.

Electronic simulation is yet very limited. A visual development environment for electronic simulation similar to LabView would be desirable. Several visual development environments exist as extensions for the chosen game engine Unity, they are not yet used for electronic prototyping, though.

Additionally, the virtual environment should provide feedback during the kinematic simulation with respect to collision detection, reachability, or kinetic plausibility, and so on.

Computer supported evaluation of the design goals should be an integral part of the design environment, but this still is a challenging task.

For increased precision of mechanical and motorized design and control, additional environments should be tested, e.g. TinkerForge and MakerBeam.

VI. SUMMARY AND OUTLOOK

The main goal of this paper was to identify important features of a rapid prototyping design environment for kinetic architecture. This was approached in a rapid prototyping fashion, itself. It started with a prototype for a design environment which employed a combination of readily available, low-cost systems, and was evaluated with a simple test case. More specific conclusions from this first approach have been summarized above. The following more general conclusions address conceptual issues.

Above all, the early stages of design need appropriate digital tool support with functional aspects. Sketching functional properties as outlined by [24] or the integration of feature modelling techniques as proposed by [25] could address this open issue.

As architecture has a strong focus on aesthetic qualities, a design environment for kinetic architecture has to target the aesthetics of motion as well. "While certain aspects of adaptive systems in architecture have little or no impact on design ... others are intimately related with design decision making. Material choices, sizing and spatial configurations are extremely important in the design and function of these systems and can have a strong impact on the aesthetic result of the design." [8]

To address the complexity of the involved challenges, the design process for kinetic architecture requires an interdisciplinary approach.

REFERENCES

[1] H. Andrä, R. Fink, "Renewing the Reichstag," *Structural engineering international* 7(4) pp.245-248, 1997.
 [2] R. Fink, K. Horstkötter, S. Zschippang, "Die neue Kuppel auf dem Reichstagsgebäude," Ernst&Sohn, *Stahlbau* 68(7), 1999.

[3] C.A. Pan, T. Jeng, "A Robotic and Kinetic Design for Interactive Architecture," in *Proc. SICE*, Taiwan, pp.1792-1796, 2010.
 [4] Milwaukee Art Museum, <http://mam.org/visit/burke-brise-soleil.php>
 [5] B. Peters, X. De Kestelier, "Rapid Prototyping and Rapid Manufacturing at Foster + Partners," in *Proc. ACADIA 08*, pp.382-389, 2008.
 [6] B. Kenzari, "Physical Modeling: the Convergence of Cutting edge Technologies and Miniature Tooling," *intl. J. architectural computing* 4(3), 2006.
 [7] C.A. Pan, T. Jeng, "Cellular Robotic Architecture," *intl. J. architectural computing* 10(3), 2012.
 [8] D. Rossi, Z. Nagy, A. Schlueter, "Adaptive Distributed Robotics for Environmental Performance, Occupant Comfort and Architectural Expression," *intl. J. architectural computing* 10(3), 2012.
 [9] P.H. Kirkegaard, I.W. Foged, "Development and Evaluation of a Responsive Building Envelope," in *Proc. Adaptive Architecture Conference*, London, University of Nottingham, 2011.
 [10] S.F. Chien, "Smart building products," in *Proc. 14th Intl. Conf. Computer Aided Architectural Design Research in Asia*, Yulin, Taiwan, 2009.
 [11] M. Senagala, "Rethinking Smart Architecture: Some Strategic Design Frameworks," *intl. J. architectural computing* 4(3), pp. 33-46, 2006.
 [12] Z. Drozdowski, "The adaptive building initiative: the functional aesthetic of adaptivity," *Architectural Design* 81(6), pp.118-123, 2011.
 [13] P. F. Jacobs, "Rapid prototyping & manufacturing: fundamentals of stereolithography," Society of Manufacturing Engineers, 1992.
 [14] A. Gomes de Sá, G. Zachmann, "Virtual reality as a tool for verification of assembly and maintenance processes," *Computers & Graphics* 23(3), pp.389-403, 1999.
 [15] P.R. Moore, J. Pu, H.C. Ng, C.B. Wong, S.K. Chong, X. Chen, J. Adolfsson, P. Olofsgård, and J.-O. Lundgren, "Virtual engineering: an integrated approach to agile manufacturing machinery design and control," *Mechatronics* 13(10), pp.1105-1121, 2003.
 [16] A. Jönsson, J. Wall, G. Broman, "A virtual machine concept for real-time simulation of machine tool dynamics," *Intl. J. Machine Tools and Manufacture* 45(7), pp.795-801, 2005.
 [17] Q. Shen, J. Gausemeier, J. Bauch, "A cooperative virtual prototyping system for mechatronic solution elements based assembly," *Advanced Engineering Informatics* 19(2), pp.169-177, 2005.
 [18] J.O. Hamblen, G.M.E. van Bekkum, "An embedded systems laboratory to support rapid prototyping of robotics and the internet of things," *IEEE Trans. Education* 56(1):121-128, 2013.
 [19] A. Parkes, H. Ishii, "Kinetic sketchup: motion prototyping in the tangible design process," in *ACM Proc. 3rd Intl. Conf. Tangible and Embedded Interaction*, pp. 367-372, 2009.
 [20] B. Hartmann, S. R. Klemmer, M. Bernstein, L. Abdulla, B. Burr, A. Robinson-Mosher, and J. Gee, "Reflective physical prototyping through integrated design, test, and analysis," in *Proc. 19th annual ACM symposium on User interface software and technology, UIST '06*, pp. 299-308, New York, USA, 2006.
 [21] J.W. Park, "Interactive Kinetic Media Facades: A Pedagogical Design System to Support an Integrated Virtual-Physical Prototyping Environment in the Design Process of Media Facades," *J. Asian Architecture and Building Engineering* 12(2), pp.237-244, 2013.
 [22] <http://www.hoberman.com/>
 [23] M. Fox, M. Kemp, *Interactive architecture*. New York: Princeton Architectural Press, 2009.
 [24] T. Shao, W. Li, K. Zhou, W. Xu, B. Guo, N. Mitra, "Interpreting Concept Sketches," *ACM Trans. Graph.* 32(4), 2013.
 [25] W.F. Bronsvort, R. Bidarra, H.A. van der Meiden, T. Tuteneel, "The increasing role of semantics in object modelling," *Computer-Aided Design and Applications* 7(3), pp. 431-440, 2010.