

Industrial Robots for Design Education: Robots as Open Interfaces beyond Fabrication

Sigrid Brell-Çokcan* and Johannes Braumann

Association for Robots in Architecture & TU Vienna, Vienna, Austria
{sigrid,johannes}@robotsinarchitecture.org

Abstract. For a long time, robotic arms have been a common sight in many industries. Now, robots are rapidly entering architectural education. Within the past few years, more than 20 architectural faculties throughout the world have set up experimental labs with one or more robotic arms. This paper will discuss the use of robots in education beyond the scope of CNC fabrication, as open interfaces that confront students with problem-solving, geometry, and programming.

Keywords: Design education, teaching robotics, robots in architecture, open interfaces, computer aided architectural design.

1 Industrial Robots in Architecture

In less than a decade, robotic arms – also referred to as industrial robots – have become a common tool in architectural education. Historically, the use of robotic labour in architecture has been explored in many ways, from illustrations of robots building houses dating back to shortly after the turn of the 20th century [1], to robotic houses, such as Archigram’s “Walking City” [2] or more recently Gregg Lynn’s “RV Prototype” [3]. Similarly, today’s use of industrial robots in architecture also covers an extremely wide spectrum, from CNC (Computer Numeric Control) fabrication machines to sensoric devices.

By design, industrial robots have always been built as multifunctional, modular machines that can be employed for a multitude of tasks, depending on which tool is attached to their flange, similar to how a human hand can hold different tools. In an architectural context, be it in education or in professional practice, this is a significant advantage, as a single robot can perform various processes that would commonly require more than one machine. This saves money, while at the same time offering a non-cubic workspace that is significantly larger than most commonly available CNC machines of a similar size. As unlike these devices, robotic arms do not have to be built “around” the workspace, they are much lighter, easier to install and can be adapted for new tasks.

These features make industrial robots ideal tools for prototypical production of 1:1 objects or prototypes in an architectural scale. It is therefore not surprising, that the

* Corresponding author.

most widely published robotic projects in architecture, are based on *scale* and *versatility*. For example, the work of Gramazio and Kohler at ETH Zurich [4] deals mostly with the use of gripping devices to create non-standard structures out of standard building materials such as bricks or wooden logs. The Institute for Computational Design at the University of Stuttgart is best known for their research pavilions, which use a robotic arm for the milling of the wooden elements [5].

Therefore, industrial robots are used as versatile fabrication tools that can switch from being a brick-stacking device to a multi-axis milling machine just by changing its end-effector. The advantages and impact that such a versatile machine can have on architectural education have not yet been discussed.

In our six years of teaching architecture students the use of industrial robots we have encountered various significant aspects of (digital) architecture that are hard to teach in general, but can be intuitively explored by having students interact with and control robotic arms. Robots are therefore not used as a pure fabrication device, but rather as an *open interface* for architectural education (Fig. 1).

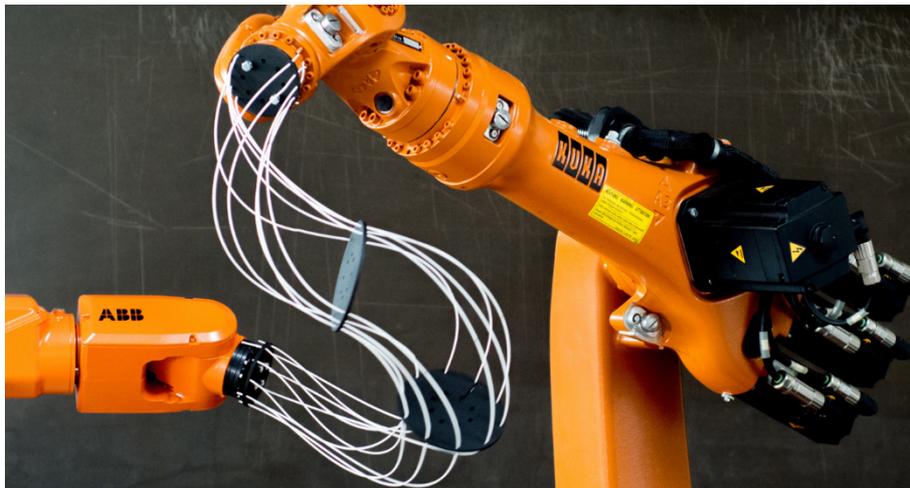


Fig. 1. Formfinding of threedimensional structures using industrial robots for education

2 Geometry and Robotics

2.1 The Offset Problem

One of the most complex tasks in CAAD (Computer Aided Architectural Design) is the translation of mathematically defined elements with zero thickness (e.g. NURBS surfaces) into physical space with both volume and thickness. In the past years, CAAD tools have made great advances and allow even new users the design of elaborate, virtual structures. However, the translation of such designs into buildable geometries is often missing in architectural education - an issue that confronts architectural offices with highly complex mathematical problems. While architects do not

necessarily have to solve such problems themselves, they have to understand the implications of geometric operations. The most common way of translating a 3D surface into a volume is a simple offset operations. As only developable surfaces can be geometrically accurately offset, expensive, non-standard geometric CAD software such as Evolute Tools or specialized geometric consultancy offices are required to compensate geometric weaknesses of CAAD tools and to make arbitrary surfaces constructible.

For education, good ways of spotting such issues in time are model-building, fabrication, and prototyping in general. In our research and teaching, we work with tool path layouts as a bottom up strategy to incorporate fabrication constraints as design parameters for surface and/or solid creation in CAAD. As an example, the fabrication strategy of flank milling is an especially valuable educational task, as it clearly demonstrates properties of developable surfaces and offset operations in physical space.

2.2 Flank Milling

Flank milling, is an advanced machining strategy where instead of the tip of the tool, the side of the cutter touches the surface and removes the stock in front of it [6]. Because of this, surfaces can be cut with a single tool path, instead of using the less efficient standard point milling methods [7]. Our research has shown that the tool-path length can be reduced to one tenth in architectural projects as well. This efficiency makes flank milling especially interesting for architectural education, when many individual projects have to be produced in a minimum of time [8]. Furthermore, flankmilling requires students to carefully consider the geometric properties and to design intelligent surfaces. In CAD, flank milling can be represented via two curves, with the tooltip following the lower base curve, and the tilting angle of the tool axis defined by the upper guide curve. Geometrically, this results in a ruled surface, with the tool axis following the surface's families of straight lines. However, this assumes that the tool's diameter is zero – something that does not exist in the physical work space. As soon as the tool exhibits a diameter, offset surfaces are created on both sides, which – as explained above – only result in accurate, ruled offsets if the original surface is not only ruled, but also developable. In the StackIt project, students attempted to create an efficient, free-formed surface out of a single panel of EPS. Tests were made by splitting the volume with the cutting surface, resulting in perfectly fitting pieces that could be stacked on top of each other. However, when the project was fabricated, students noticed that the parts had no perfect fit which, if scaled up to 1:1, would result in significant deviations between the resulting edge curves. Further investigation showed them that the reason for that behaviour is that when the tool's angle changes, so does the cross-section of the tool diameter when seen from above (Fig. 2).

One solution to solve this problem was achieved by milling tangentially along the surface. While this is still not a geometrically perfect solution - unless the cutting surface is developable - the results were within tolerances and acceptable for architectural applications. Understanding the geometric properties of offset, developability,

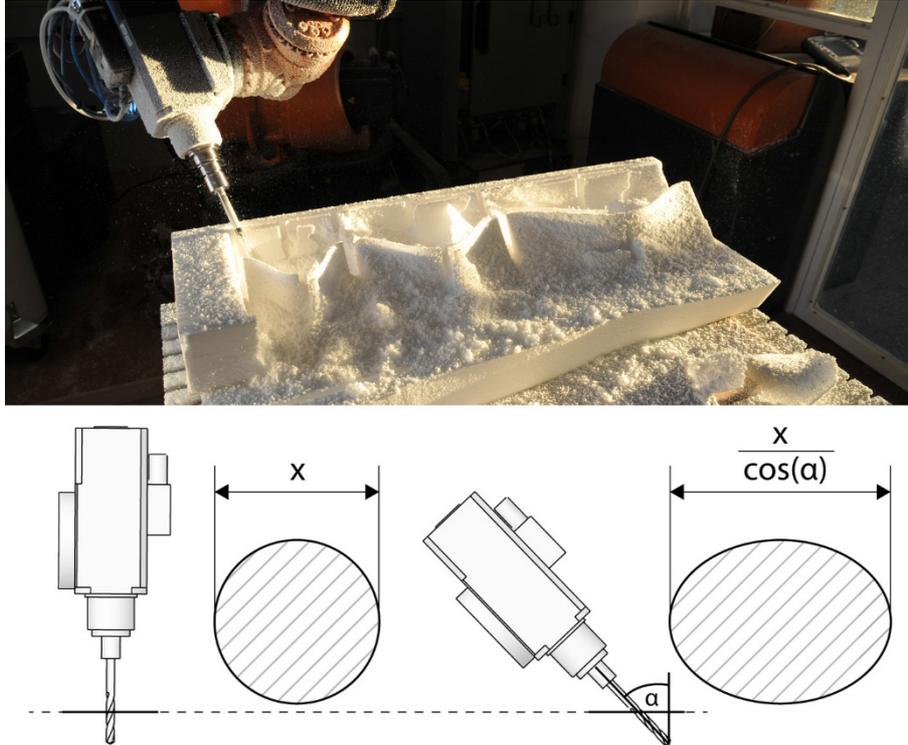


Fig. 2. Flank-milling complex, geometric structures (above), geometric implications of flank milling (below)

and ruled surfaces gives students knowledge that is applicable in many scales and contexts, from metal façade designs (refer to work by Gehry [9]) to wire-cut, ruled, EPS-formwork for concrete [10]. We refer to this teaching method as material operability of design.

2.3 Complex Systems and Degrees of Freedom

Robots lend themselves very well as a pedagogical tool to spot the difference between CAD curves and actual physical tool paths and to explain complex machinic processes, because students cannot immediately grasp the complexity of their kinematic movements.

An industrial robot is a serial articulated arm with six degrees of freedom that can even be enhanced by additional external axes, such as a linear rail, a turntable or even cooperating robots, making them significantly more complex than regular three or five axis milling machines. We regard the handling of complex systems and degrees of freedom (DOF) as a very important part of architectural education.

Robot kinematics clearly show students the perceived “unpredictability” of machinic movements. When a human is instructed to move his arm from one point to

another in the most efficient manner, he will most probably move it in a straight line. However, an industrial robot will choose the most economic movement that requires the least amount of axis rotations, which is hardly ever a straight line. While this behaviour is perfectly logical, it is difficult for a viewer to intuitively predict the robot's trajectory. Understanding such concepts is highly important for machinic processes where intuitive human thinking and mathematic logic do not overlap.

3 Teaching Programming

Due to the complexity of architectural designs, geometry in CAD is tightly linked to programming. Recent developments such as the graphical algorithm editor Grasshopper by David Rutten enable designers to use visual programming for defining complex, geometrical algorithms and apply them automatically on surfaces and other geometries.

3.1 New Programming Interfaces

The dependence of CAM on preset fabrication strategies prompted us to develop our own robot software tool KUKAlprc (parametric robot control) [11] for the visual programming environment Grasshopper, which enables an accurate, kinematic simulation of robot toolpaths, along with the immediate conversion of toolpaths into native robot code, to be executed directly at the robot. Such software greatly facilitates robot programming in architectural education, as students no longer have to learn different programming environments, but can focus on one CAD software – in this case Rhinoceros and the accompanying parametric modelling environment Grasshopper. The development of these tools had a liberating effect on our teaching, as we could pose tasks that no longer depended on e.g. a certain milling strategy. Instead, students were tasked with creating their own parametric design tools that not only allow design iteration, exploration, and optimization, but by including the robot-components also enable the inclusion of robot simulation – to which we refer to as production-immanent design.

While it can be argued that architects do not need in-depth knowledge of CAM software, parametric modelling for design exploration or the automation of repetitive tasks has become a sought-after skill in architectural offices.

3.2 Programming Complex Processes

An additional aspect that is both relevant to architecture, visual programming, and robotics, is the topic of complex processes. It is extremely important for an architect to understand and gain knowledge to plan and control the sequence of assembly

While humans can be given vague instructions, this is not possible with machines. Even a simple pick-and-place operation, where a robot picks up a part and places it somewhere else consists of a surprising amount of actions: a) move into safety distance above item, b) move down, c) close gripper, d) move up to safety plane, e) move

towards new position, f) move down, g) open gripper and release item. An even greater challenge that has so far not been sufficiently explored is the manipulation of non-rigid materials in machinic processes. In our research and research-oriented teaching, we have investigated the robotic forming of splineTEX, a composite material that is provided as flexible tubes, which can be hardened via epoxy infusion. By approximating the material behaviour in a custom physics engine, and merging it with the robot kinematic solver, we enabled our students to intuitively explore not only the performance of the robot, but also of the material itself (Fig. 3). Visual programming allows them to check every single fabrication step, instead of just running a series of text-based commands. This was applied in a student project, where a flexible frame with a membrane was mounted onto three industrial robots, which – by moving – manipulated and changed the membrane’s form – a process that students were able to approximate using the Kangaroo physics solver in Grasshopper.



Fig. 3. Research oriented teaching: Student projects exploring robotic interaction with tube structures (above), research project on robotic winding for automotive applications of splineTEX, simulated and programmed using Grasshopper and KUKA|prc

Compared to regular, text-based programming, visual programming greatly facilitates working with complex processes, as each process can be visualized and its order shifted, simply by plugging and unplugging its connection to the component that generates robot code.

3.3 New Interfaces for Robotic Programming

An area of research that is highly relevant not only to the use of robots in architecture, but to robotic fabrication in general, are new interfaces. We have developed and prototyped several interfaces that allow students to interact with industrial robots, without having to deal with the intricacies of the control panel interfaces. Especially at the beginning of a course, it is highly important that students get an intuitive feeling for robotic fabrication. To a certain degree, this can be achieved via robot simulation in a virtual environment (see Section 3), but is more efficient when performed in physical space. The developed interfaces were mostly based on strategies that are common in electronic entertainment, such as a touch-based user interfaces that not only allows the real-time adjustment of the robot's position, but additionally shows the robot in 3D directly on-screen. Similarly, a gesture-based control interface was tested using a Microsoft Kinect motion capture device, which allowed the students to interact with the robot via their body's movements. As mentioned before, the scope of such experiments is not limited to allowing architecture students to interact with complex, kinematic machines, but was performed as part of a larger research project that involves the topic of gender equivalence, technology interest of children and the demographic change (Fig. 4). With a decreasing working population, it will be necessary for the European industry to increase robotic fabrication to be able to keep up with demand. As skilled labour will soon be less available than today, such interfaces are expected to allow people without an in-depth robotic knowledge to collaborate with industrial robots [12].



Fig. 4. New robotic interfaces for intuitive robot control - enabling even children to interact with and “program” robotic arms – are explored in the context of demographic change

4 Creating Custom Fabrication Tools

The custom software tools in Section 3.2 enable students and researchers to go beyond standard pathplanning for e.g. milling, to create new, parametric fabrication strategies in an architectural context. However, to apply these strategies outside of a virtual environment, the prototyping of custom end-effectors is required.

4.1 New, Affordable Robotic Tools

In industry, the development and integration of new robotic end-effectors is often more expensive than the robotic arm itself. As such a budget is not available for prototypical fabrication strategies in architecture, the design and integration of custom end-effectors has become an integral part of our robotic courses. The resulting output varied greatly in complexity, from fully functional gripping devices that are 3D-printed, via attachments that themselves held other tools such as pens, laser-pointers, and cameras, to simple end-effectors that consist of a base plate and four metal pins, for experimenting with winding patterns.

Integrating end-effectors that have different states, e.g. a switchable gripper or spraygun, requires custom electronics. Not long ago, such operations would have required the services of an electrical engineer. With platforms such as Arduino, that merge a standardized microcontroller with an accessible programming language, even architecture students can program their own devices that can receive commands from the parametric design environment, or even from the robot itself.

5 Conclusion

In the past few years, the use of robots in architecture has taken great strides. It has been our experience that learning to handle such a machine is a process that usually takes several years. Now, there are more and more people who can not only program robots using software provided by the industry, but design their own software tools for robotic fabrication outside the scope of traditional CNC application.

For us, advancing from being only “users” of industrial robots had a significant impact on our approach of teaching industrial robots to architecture students: Creating any aesthetic output using robotic fabrication is not the goal anymore, but to use industrial robots as open design interfaces with adaptable tools to achieve a goal. In the Architectural Geometry course, which is offered in cooperation with TU Vienna’s department for industrial geometry [13] to Master-level architecture students, the properties of geometric entities such as coordinate systems and ruled surfaces are first explained by mathematicians, and then actively applied in Grasshopper and used for fabrication using a KUKA robot. This leads to the advantage that students understand that they are not learning about geometry to succeed in an exam, but to actively apply it in a later stage e.g. with a robot. The robot therefore acts as a motivator and proof of concept.

The design of custom fabrication processes in our design to production lecture was also well received by students, as they can adapt their projects according to their individual strengths, e.g. by scripting highly complex programs or rather building elaborate end-effectors. This has led to a similarly diverse output, from workflows that capture the movements of a dancer and transform it into robot toolpaths, to adaptable structures used for form exploration that are held by cooperating robots.

We argue that industrial robots offer a great potential in architectural education, as their multifunctionality allows them to be used far outside the scope of regular CNC fabrication. While the handling of large robots requires constant supervision and has to be done in dedicated robotic cells, many universities are now opting to purchase additional desktop-size robot that allow students to work independently. As these machines are programmed the same way as the larger robots the results can be immediately applied at the large machines for full scale applications.

Machinic processes are becoming increasingly important in both architectural education as well as in professional offices. While it is not necessary for every architectural graduate to have in-depth knowledge of CAM, robotic arms can act as open interfaces that transfer valuable knowledge from many disciplines such as geometry and programming.

However, with the increasing proliferation of robotic arms in the construction industry, we believe it to be very probable that many of today's students will sooner or later be directly confronted with robotic fabrication in their professional practise.

Acknowledgements. We want to thank our students for their enthusiasm and hard work. Projects by Marco Tomicic (Fig. 1), Müller & Vladikov (Fig. 2), and Kris Hammerberg (Fig. 3 above) are presented. This research has been supported by the Austrian Academy of Sciences (grant nr. 273458) and by the Austrian Research Agency (FFG, grant nr. 835992).

References

1. Villemard: Chantier de construction électrique, Bibliothèque nationale de France, Paris (1910)
2. Cook, P.: Archigram. Princeton Architectural Press, New York (1999)
3. Lynn, G.: RV (Room Vehicle), <http://www.glforn.com>
4. Gramazio, F., Kohler, M.: Digital Materiality in Architecture. Publishers, Baden (2007)
5. Menges, A.: Morphospaces of Robotic Fabrication. In: Brell-Cokcan, S., Braumann, J. (eds.) RoblArch: Robotic Fabrication in Architecture, Art, and Design, pp. 28–47. Springer, Vienna (2012)
6. Li, C., Bedi, S., Mann, S.: Surface Design in 5-axis Machining. In: Proceedings of the International Conference on Smart Machining Systems, Gaithersburg (2007)
7. Waldt, N.: NC-Programmierung für das fünfsachsige Flankenfräsen von Freiformflächen. Universität Hamburg, Hamburg (2005) (in German)
8. Brell-Cokcan, S., Reis, M., Schmiedhofer, H., Braumann, J.: Digital Design to Digital Production: Flank Milling with a 7-Axis Robot and Parametric Design. In: Proceedings of the 27th eCAADe Conference, Istanbul, pp. 323–330 (2009)
9. Shelden, D.: Digital surface representation and the constructibility of Gehry's architecture. Massachusetts Institute of Technology, Cambridge (2002)
10. Feringa, J., Sondergaard, A.: Design and Fabrication of Topologically Optimized Structures. In: Proceedings of the 30th eCAADe Conference, Prague, vol. 2, pp. 495–500 (2012)
11. Braumann, J., Brell-Cokcan, S.: Parametric Robot Control: Integrated CAD/CAM for Architectural Design. In: 31st Annual Conference of the Association for Computer Aided Design in Architecture, Calgary, pp. 242–251 (2011)
12. Petereit, S., Albert, A., Jeridi, N., Schoenleber, R., Rebmann, C., Vallery, H.: Work Life in the Light of the Demographic Change: Case Study Force Assistive Device for Craftsmen. In: Wichert, R., Eberhardt, B. (eds.) Ambient Assisted Living. Advanced Technologies and Societal Change, vol. 2, pp. 45–60. Springer, Heidelberg (2012)
13. Pottmann, H., Asperl, A., Hofer, M., Kilian, A.: Architectural Geometry. Bentley Institute Press (2007)