Galaxies of Non-/ Human Thinking: Opening Mind – Noticing Kind

Johannes Braumann

The digital transformation is no longer the domain of computer scientists and programmers alone; it is affecting all of our daily lives and therefore calls for education at all levels of society. However, digitization is frequently only explored and explained in a virtual context; processes of physicalizing data, of digital fabrication, are often omitted. Closely managing the transition from the digital back to the physical is especially important for people who want to realize custom processes and products that go beyond the established, standardized methods of industry, such as those working in creative fields: artists, architects, designers, craftspeople, and hybrid professionals. Rather than try to teach these diverse groups to build their own machines, industrial robots can be repurposed as a way of introducing individualized digital fabrication to new users, thus fostering positive change through experimentation and innovation. People are not replaced, but equipped with new digital and physical tools that allow them to go beyond the state of the art.

Building Data – Between Expectations and Reality

Popular media often depict robots as intelligent, even humanoid machines that can interact naturally with people. A current example of that is Boston Dynamics' Spot (Fig. 1), an impressively engineered robot dog that has been shown to dance, navigate autonomously through difficult terrain, or simply replace its animal counterpart by walking on a leash. Yet a closer look at the technologies behind those projects reveals dance routines that are merely pre-programmed and tested rather than organically improvised. The navigation relies on complex, external processing power to map and interpret environments and, ironically, it's not usually the person who walks the dog, but the dog who walks the person. Needless to say, attaining the level of interaction and performance seen in the media requires not just expensive hardware, but in-depth, multidisciplinary knowledge as well.

Building robots in an educational setting has long demanded a high-level understanding of mechanical engineering, electronics, control theory, computer science, mathematics and more. The rise of physical computing platforms like Arduino (Fig. 2a) has meant not only widespread availability of affordable, open-source hardware and microcontrollers, but most importantly accessible software environments that are supported by large communities of both beginners and experts. And yet, even with those platforms – which greatly abstract the complexities of addressing sensors and controlling motors – a non-expert's results are bound to fall far short of anything the media portrays robots as capable of doing. Moreover, these systems are constrained in size, as large-scale, heavy-duty systems involve high voltage and are too heavy to be supported by cheap, 3D-printed elements. The wide gap between expectations and reality presents significant challenges when teaching robotics to users who are not content to learn something for the sake of gaining knowledge, but instead want to actively apply robotic technologies as a way of facilitating their professional practice. Examples might include a carpenter in need of a system to support the lifting of heavy loads, for instance; an architect interested in using hot-wire cutting to extract complex shapes from foam materials; or a dancer whose envisioned stage piece involves interaction with a human-scale machine, with both human and non-human elements noticing the other's kind.

Introducing Industrial Robots

Creating a custom machine that can perform these tasks would be highly challenging. There are, however, solutions to be found in the realm of industry, where industrial robots and robotic arms manipulate heavy loads over large distances with a high degree of accuracy (Fig. 2b). Their origins are the same as those of computer numerical control (CNC) machines: in 1952, researchers at the MIT Servomechanisms Laboratory converted a commercially available milling machine into a device with three-axis simultaneous movement.[1] Two years later, George C. Devol filed his patent for a "general purpose machine that has universal application to a vast diversity of applications where cyclic

1 Cf. Parsons & Stulen, 1958.

control is to be desired"[2]; the result was the Unimate, the first industrial robot, a prototype of which was sold to General Motors in 1961. Less than a decade later, in 1969, Victor Scheinman presented the Stanford Arm, a robotic manipulator with six degrees of freedom. Driven by electric motors, the robotic arm displayed a range of motions approaching the fine motor skills of a human arm.[3]

A fundamental difference between a CNC machine and an industrial robot is that the latter is not application-specific, but (as per Devol's patent) a general-purpose machine that can be used to perform a variety of very different tasks. It is comparable to a human arm with extended functionality, infinitely increased by the use of external tools. The concept of the industrial robot has since been further iterated based on user requirements enabling new ways of fabricating and assembling, in particular for the automotive industry, which – with 125,700 robots installed per year – is still the primary user of robotic arms.[4] Current-generation robotic arms, such as KUKA's Quantec-2 series, weigh 1.1t and consist of six axes that can move a payload of up to 300kg (661 lbs) at a maximum 260 degrees/second over a reach of 2.7m (8.85 ft), with a repeatability speed of ± 0.05 mm.[5]

The appeal of industrial robots for large-scale mass manufacturing is that a single platform can be used for a wide array of applications. Rather than having to build multiple custom machines, a single type of machine can deploy a number of tools to perform any number of tasks. While in most cases a generic machine will not attain the kind of performance expected of a specialized machine, mass-produced robotic arms are designed to be highly reliable and offer a known level of performance. Furthermore, at a price range of around 30,000–70,000 euros, industrial robots are more affordable than many other CNC machines and are available in large quantities on the used market. These advantages make robotic arms interesting to fields beyond the automotive sector, such as the creative industries.

However, a remaining challenge is the programming. In industry, robotic arms are mostly programmed once, and then perform that same, single action or sequence of actions over their entire lifespan. This makes it possible to dedicate a large amount of time to programming, as small improvements

2 Devol, 1961.
3 Cf. Feldman et al., 1969.
4 Cf. IFR, 2018.
5 Cf. ISO, 1998; KUKA, 2019.

will lead to significant time savings over the years. 'Online programming,' also known as 'teaching,' is the most common way to program robots in industry. In this case, the programming happens entirely over the robot's teach pendant (Fig. 3) – often a tablet-like controller – where the user can save the robot's current position in a file and then repeat that process to generate a toolpath.[6]

The process is time-consuming, and requires the user to think within the robot's coordinate systems as they push the relevant button on the teach pendent to move the machine. It can be streamlined by grabbing the tool of a compliant robot and moving it by hand, or external tools that allow the user to rapidly capture positions without having to move the robot to those positions.[7] Whereas the automotive sector builds on assembly techniques with robots performing single, specific fabrication steps on a production line, newer fields of application for robotic arms take advantage of the robot's flexibility; a single robot can be programmed to perform a wide variety of tasks depending on the production process at hand. As such, efficiency of the robotic program is not a high priority, as tasks are performed irregularly and with longer pauses in between while other production steps are performed by humans.

That said, task definition itself must be done in a rapid and (semi)automated way. Manually moving the robot to each position may take more time than a human would need to simply perform the desired task by hand.

Beyond Industry

Given these considerations, there is a great need for environments that make the programming of industrial robots more accessible to a broad array of users. A common approach is to use flow-based programming, a paradigm that works with visual 'blocks' rather than text.[8] The outcome is often a flowchart-like program wherein modules representing certain functionality or processes are connected via lines or arrows that define their parametric relationships.

In industry, environments such as drag&bot[9], Fox|Core, and Franka Emika's software Desk allow the visual programming of industrial robots via web browser. Within the greater area of robotic research, there are also educational

6 Cf. Pan et al., 2012.
7 Cf. Müller, Deuerlein & Koch, 2021.
8 Cf. Morrison, 2010.
9 Cf. Naumann et al., 2006.

environments such as Scratch[10] and Blockly, a block-based visual programming language that has been expanded for use on industrial robots.[11]

However, these environments have mostly been developed for industrial use cases such as pick-and-place processes, where robotic manipulation occurs at the end of a development process and is usually unable to feed back information. By contrast, an ideal robotic fabrication workflow for digital artists and designers would be tightly integrated into the design process itself – something that generally happens within computer-aided design (CAD) software. An intelligent system keeps track of changes to geometry, synchronizing these in simulations of robotic fabrication in a way that directly informs the design process – a concept known as production-immanent design (Fig. 4).

Using robotic arms abstracts the complexities of designing an entire machine. In other words, rather than having to deal with mechanical, electrical, and programming challenges, robotic arms allow users to focus entirely on movement. A single line of code can tell the robot to move to a certain position within its workspace while it internally calculates the speed, acceleration, deceleration, and other factors of each of its six motors – all with the aim of reaching that position with a high degree of accuracy in the shortest possible time. As digital artists and designers work within CAD, an approach is needed to link the movement of the robot to its geometry as directly as possible (Fig. 5).

When introducing innovations such as new technologies, it is of crucial importance to consider the individual needs of a user group. Rather than create separate tools for linking design and robotic fabrication, the author of this essay and Sigrid Brell-Cokcan began to develop plugins for Grasshopper,[12] a flow-based visual programming environment that is tightly integrated with Rhinoceros, the CAD software developed by McNeel. The resulting software, known as KUKA|prc[13] (Fig. 6), is currently being used by more than 100 universities around the world; over 50 companies use it for such non-standard, innovative production processes as large-scale 3D printing (Branch Technology and Aeditive), carbon fiber lay-up (Carbon Axis), and timber constructions (Züblin Timber). Even large companies like Boeing and Adidas have employed

10 Cf. Resnick et al., 2009.
11 Cf. Mateo et al., 2014; Trower & Gray, 2015; Weintrop et al., 2018.
12 Cf. Rutten, 2020.
13 Cf. Braumann & Brell-Cokcan, 2011.

KUKA|prc and are actively looking for researchers with both a creative and a technical background.

Such tools show that the rather strict traditional separation between designers and fabrication engineers has given way to a more even, collaborative environment that fosters innovation. It also encourages designers to consider fabrication-related parameters when implementing their designs, thus greatly facilitating the process.[14] This development can be viewed as part of the so-called "low-code" trend,[15] a tendency that brings new digital technologies to a wider range of users while also allowing existing industries to prototype and develop at a fast pace.

Teaching Robotics

Having this set of tools to abstract the complexities of interfacing and connecting data between design and fabrication enables a key step when teaching robotics to artists, architects, and designers: the focus shifts to the fabrication process, and away from the robot per se. In other words, the robot is considered not as an end in itself, but as an especially multifunctional tool. This is in a stark contrast to how robotics is taught in engineering classes, which center on understanding the technical framework of inverse and forward kinematics, control algorithm, and interfacing with programmable logic controllers, while the actual robotic process is often little more than an afterthought or proof-of-concept.

Of course, the physical realities of using a robot must not be ignored. In order to achieve a reliable simulation, it is important to synchronize the virtual environment with the actual, physical one. That means, on the one hand, that both simulated and physical robots need to use the same tool coordinate system (defining the offset from the flange to the tool center point, e.g. the tip of a mounted pen) and base coordinate system (defining the location of the CAD-zero point) within the robot's workspace. In addition, any obstacles such as tables and walls need to be drawn in the simulation in order to check for collisions. Once that synchronization has been performed by the user, programs should behave identically in the virtual and real environments. Data exchange between the robot and the flow-based programming environment can then happen in one of two ways: either a text file with movement com-

14 Cf. Braumann & Brell-Cokcan, 2015.
15 Cf. Fryling, 2019.

mands formatted in KUKA Robot Language (KRL) is copied to the robot controller via a flash drive or shared folder, or the robot can be controlled in real time via an industrial communication protocol.[16]

However, the program needs to be geometrically defined first. At a basic level, a robotic arm is not aware of any kind of 'actions' (picking up an object, for example). Instead, such a process needs to be broken down into its basic components. For instance, the robot is instructed to move above the object in a straight line, close the gripper, wait for a few hundred milliseconds until the gripper is closed, then move up again and to a point above the drop-off position (Fig. 7). Fortunately, visual programming allows the user to create such a sequence of events very quickly and easily re-use it: by providing multiple target positions, the system will create multiple pick-and-place operations based on the initial logic. The basic complexity of a robotic program is therefore low enough that a completely new user with basic CAD experience can create their own within 1–2 hours. At the same time, the complexity can be increased to any level, for example towards defining complex, non-planar, spatial 3D-printing processes that go beyond what is currently possible with commercial slicer software.

Simulation and Expertise

The provided simulation primarily checks whether or not the robot is physically capable of reaching a given position and if it collides with its environment. Importantly, KUKA|prc simulations do not show modeling of materials behavior, as reliable finite element modeling (FEM) and multi-physics simulation of a milling process requires both complex, expensive software and vast amounts of computational power. A current example is an application developed for a local SME specializing in polishing high-end metal molds for the automotive industry (Fig. 8a–b). The polishing toolpath depends heavily on the shape and material of the mounted polishing tool and the surface to be polished: choosing the right parameters requires a great deal of expertise, the result of years of experience with polishing.

Rather than try to duplicate that knowledge with a multi-year research project (resulting in a fully-automated process), researchers opted to create a much simpler program within a few weeks. The result exposes a number of inputs to the material experts, who can then choose the relevant parameters

16 Cf. Braumann & Brell-Cokcan, 2015.

based on their experience with the material. An immediate preview allows them to verify and validate the process and send it to the robotic arm. As such, predicting the material behavior is still the domain of the process expert, who – in the case of 'creative' applications – is usually the robot programmer him- or herself. This serves to reinforce the importance of having a dual skillset: programming skill is not per se sufficient when it comes to robotics; one also needs to be aware of the specific physical process that robot is performing.

Collaboration and Knowledge Transfer

When teaching robotics to new users, the approach at Creative Robotics is to encourage both students and professionals to pick a project that relates to their artistic or professional practice and to evaluate how it could either improve upon an existing process or enable a completely new process that would otherwise not be possible. Using a robot should never be an end in itself, but should rather create added value that goes beyond incremental changes. Abstracting the complexities of working with a machine enables a transfer of 'universal' skills that are not limited to a certain type or brand of robot. This is especially true in higher education, where only a fraction of the students taking robotics courses will actually use robots in their professional career. It also makes it particularly important to focus on actual processes rather than the underlying algorithms, as the acquired skills can then much more easily be applied in a new context.

Robotics therefore serves as a motivational tool, exposing students to flow-based programming and with that, to algorithmic thinking. The definition of toolpaths requires students to brush up on their knowledge of geometry, since terms such as 'coordinate system,' 'normal vector,' 'tangent,' and 'transformation' are crucial to achieving a level of control over the robot's trajectory. Similarly important is the ability to break down processes to their basic components, to analyze them and then build upon that knowledge. Finally, use of robotics can be a valuable interface for interdisciplinary collaboration between students of different fields of study, as it enables them to cooperate within a shared environment that is new to all involved, but still individually relevant to their work.

Two experimental, interdisciplinary formats are currently under development. Creative Robotics and the LIT Robopsychology Lab[17] aim to create a

17 Cf. LIT Robopsychology Lab, 2021.

course structure that integrates art and design students as well as psychology students, with the goal of exploring the relationship between humans and robotic arms through actual prototypes that students develop themselves using those machines. Another format by the Ars Electronica Center and Creative Robotics includes a full-day workshop to introduce law students to the wider topic of digitization and robotics, encouraging them to engage with the machines in a hands-on way that allows for closer understanding of highly relevant legal questions within the field of robotics and digitization, particularly with regard to safety and liability.

Creative Hubs

A key part of facilitating the introduction of new technologies to new groups of users is having an inspiring environment that allows people to experience technology up close. Anecdotally, universities are often seen as distant from the realities of daily life, making it challenging to reach out to new groups of users. For that reason, Creative Robotics has partnered closely with the Grand Garage, now Europe's largest maker space,[18] as it provides a highly inclusive environment that targets not only makers, but also companies and startups, primary- and secondary school students. It is also home to a number of continuing education and reorientation programs, such as those for unemployed youth.

Grand Garage itself is tied to Tabakfabrik Linz,[19] a former tobacco factory that has become a creative hub for both academic and commercial institutions. Research groups such as the University of Art and Design Linz's Fashion & Technology, Tangible Music Lab, and Creative Robotics work alongside incubators, co-working spaces, startups, established companies and a secondary school with a focus on digital humanism (Fig. 10). All together, they form a very unique environment that is particularly well-suited to a city like Linz, with its industrial heritage and current focus on digitization.

This interdisciplinary environment has already given rise to such academia-born startups as YOKAI Studios,[20] a company and collective founded by Michael Wieser and Viktor Weichselbaumer, two students in the university's Fashion & Technology program who wanted to explore additive fabri-

18 Cf. Fab Foundation, 2021.
19 Cf. Tabakfabrik, 2021.
20 Cf. YOKAI Studios, 2021.

cation technologies not to build up objects layer-by-layer, but as a means of connecting and decorating fabrics. The project was recently awarded a grant by the EU-funded Re-FREAM program and was accepted into tech2b, an accelerator program at Tabakfabrik Linz. YOKAI Studios uses the infrastructure provided by Grand Garage and Creative Robotics to advance the technological side of their product.

Conclusion

For all the association with industry, there is a great potential in the use of robotic arms by artists, architects, and designers. Startups such as YOKAI Studios and Branch Technology[21] – but also equestrian saddlemakers at Sattlerei Niedersüß,[22] stone masons at Bamberger Natursteinwerke,[23] and visual artists Charles Aweida[24] and Davide Quayola[25] – use similar tools to work toward extremely dissimilar goals, from experimentation to fabrication and performance.

While industrial automation in Europe has nearly plateaued, the use of robots will enable creative sectors to realize entirely new processes that provide a deep level of customization, allowing them to go beyond what is mass-produced in low-wage countries at a competitive price level. Given the lack of skilled labor in these fields, deployment of robots will not replace human workers, but foster upskilling, just as the use of high tech will make traditional crafts more interesting to the next generation of artisans and craftspeople. Beyond digital fabrication, robotic arms can also become an expressive tool for artists and performers (Fig. 9–11) and a powerful interface for teaching algorithmic thinking and geometry.[26] While it may not be possible to match the entertainment industry's depiction of robots in the foreseeable future – and this despite great advances in the field of artificial intelligence and machine learning – having creative users work with robotic arms lends credence to the claim that the digital transformation is a universal concern, encompassing everything from the digital to the physical.

21 Cf. Branch Technology, 2021.
22 Cf. Sattlerei Niedersüß, 2021.
23 Cf. Bamberger Natursteinwerke, 2021.
24 Cf. Aweida, 2021.
25 Cf. Quayola, 2021.
26 Cf. Braumann & Singline, 2021.

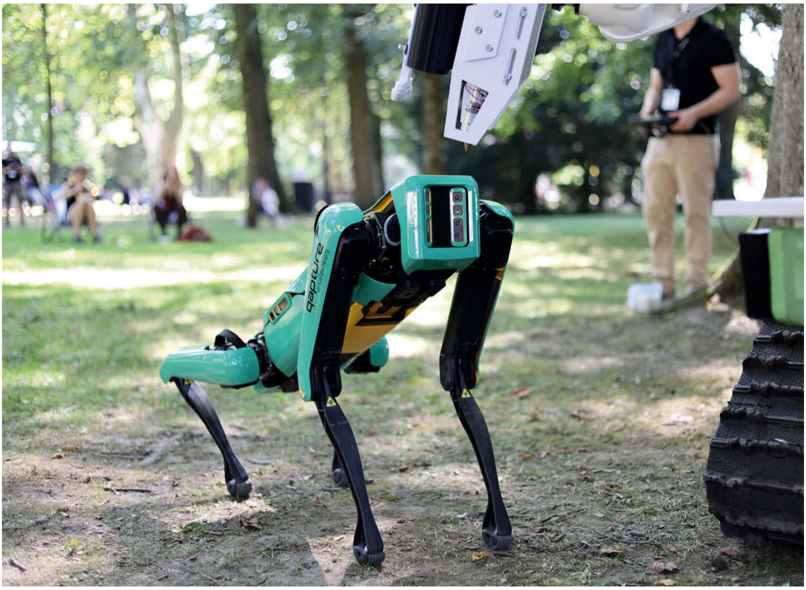

Fig. 1

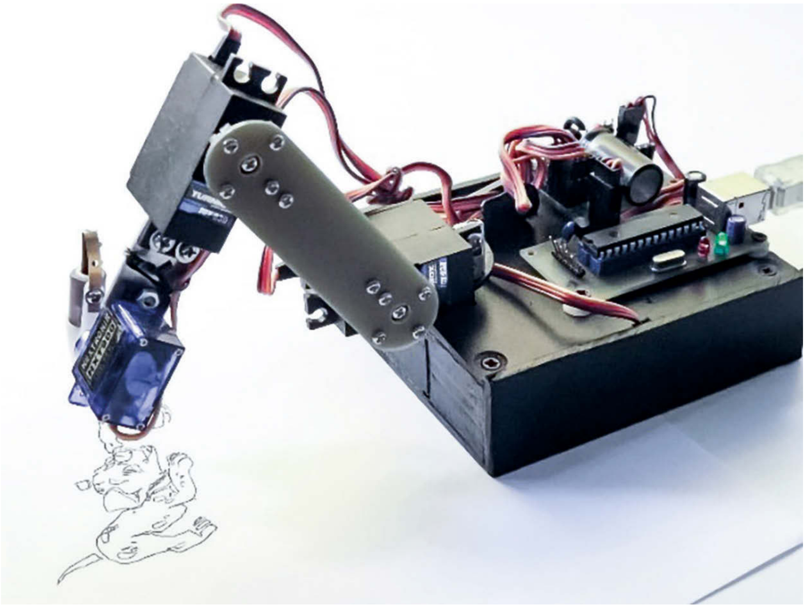

Fig. 2a

Fig. 2b

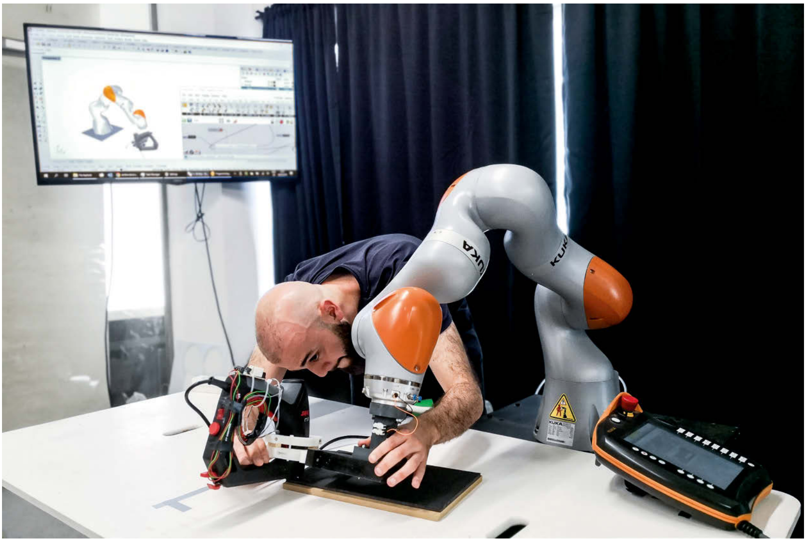

Fig. 3

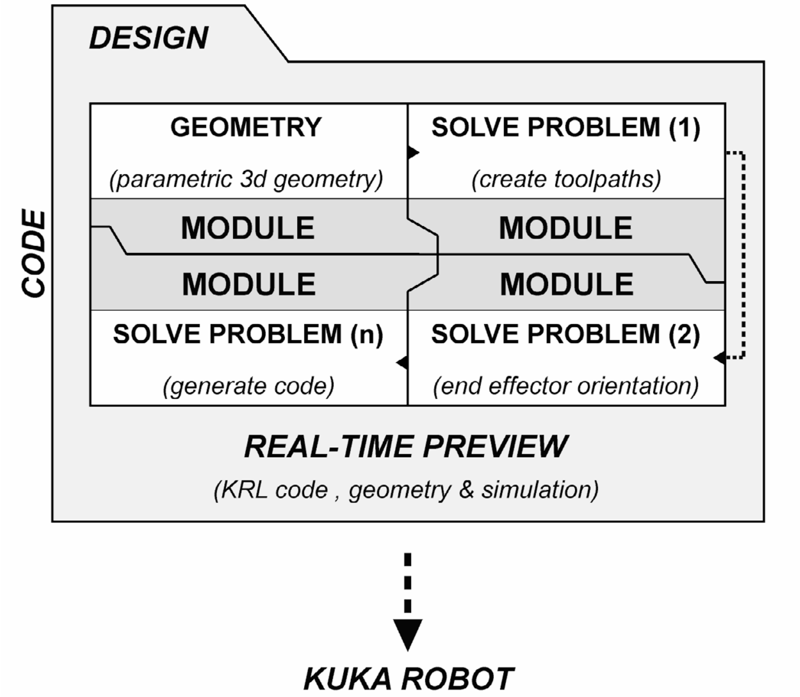
DESIGN
CODE
GEOMETRY
(parametric 3d geometry)
SOLVE PROBLEM (1)
(create toolpaths)
MODULE
MODULE
MODULE
MODULE
SOLVE PROBLEM (n)
(generate code)
SOLVE PROBLEM (2)
(end effector orientation)
REAL-TIME PREVIEW
(KRL code , geometry & simulation)
KUKA ROBOT

Fig. 4

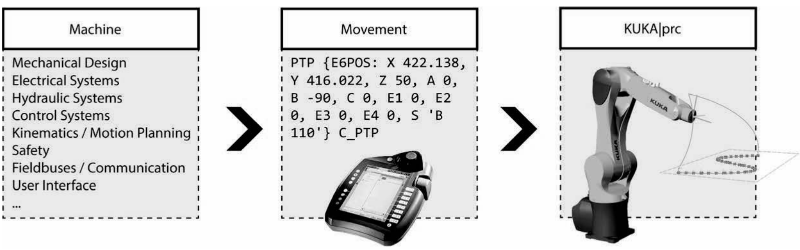
Machine
Mechanical Design
Electrical Systems
Hydraulic Systems
Control Systems
Kinematics / Motion Planning
Safety
Fieldbuses / Communication
User Interface
...
Movement
PTP {E6POS: X 422.138, Y 416.022, Z 50, A 0, B -90, C 0, E1 0, E2 0, E3 0, E4 0, S 'B 110'} C_PTP
KUKA|prc

Fig. 5

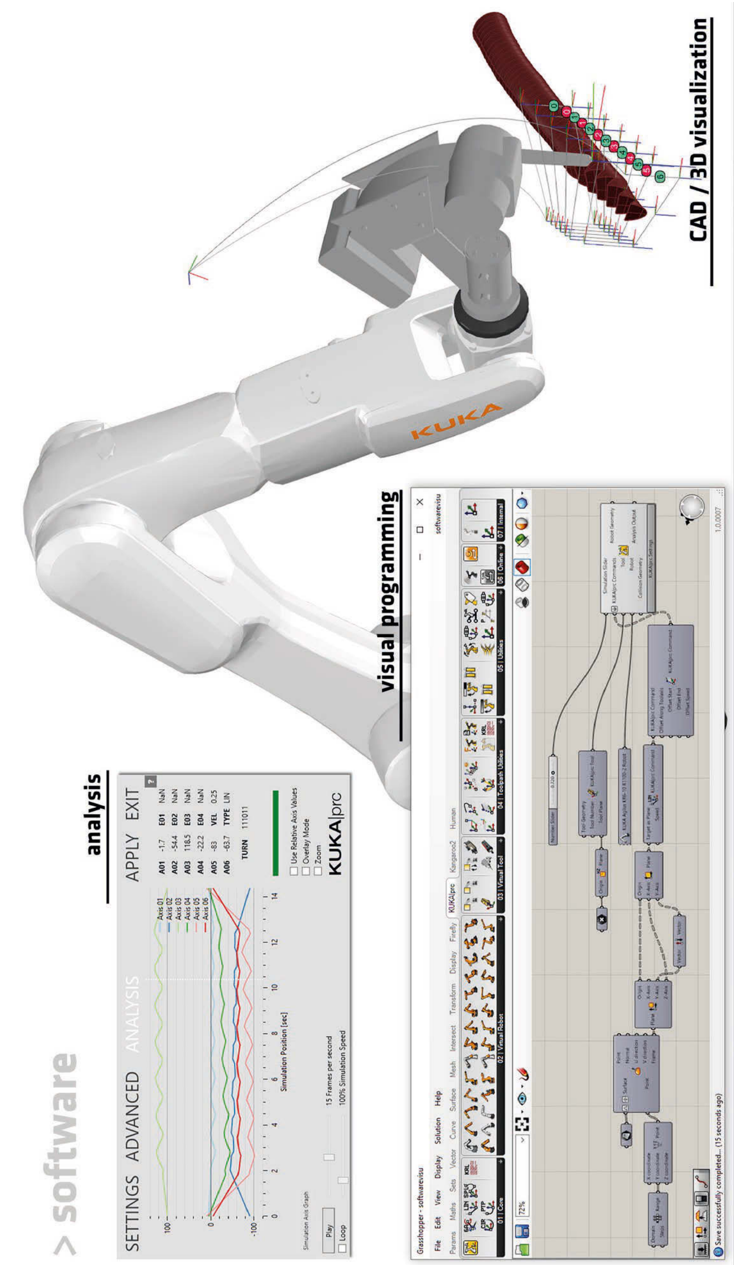
CAD / 3D visualization
visual programming
analysis
> software
KUKA|prc

Fig. 6

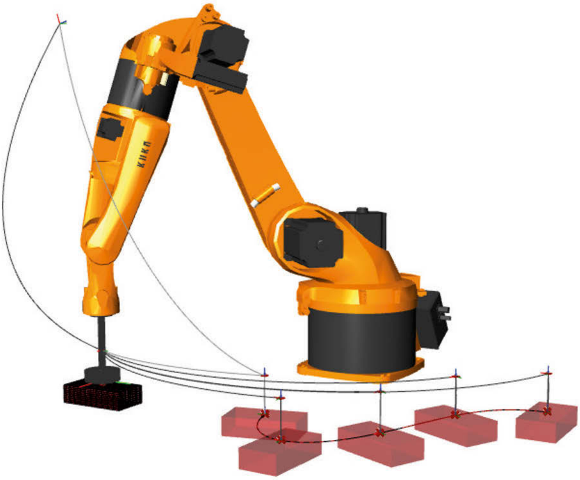

Fig. 7

Fig. 8a

Fig. 8b

Fig. 9

Fig. 10

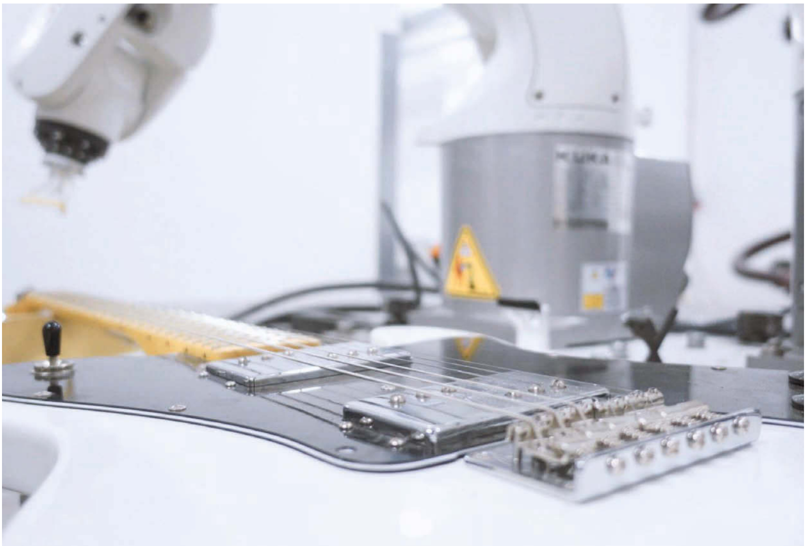

Fig. 11

References

Braumann, J. & Brell-Cokcan, S. (2011). Parametric Robot Control. Integrated CAD/CAM for Architectural Design. *Proceedings of the 31st Annual Conference of the Association for Computer Aided Design in Architecture*, 242–251.

Braumann, J. & Brell-Cokcan, S. (2015). Adaptive Robot Control: New Parametric Workflows Directly from Design to KUKA Robots. *Real Time – Proceedings of the 33rd ECAADe Conference,* 243–250.

Braumann, J. & Singline, K. (2021). Towards Real-Time Interaction with Industrial Robots in the Creative Industries. *2021 IEEE International Conference on Robotics and Automation.* 2021 IEEE International Conference on Robotics and Automation.

Devol, J. G. C. (1961). *Programmed article transfer* (United States Patent No. US2988237A).

Feldman, J. A., Feldman, G. M., Falk, G., Grape, G., Pearlman, J., Sobel, I. & Tenebaum, J. M. (1969). The Stanford Hand-Eye Project. *Proceedings of the 1st International Joint Conference on Artificial Intelligence,* 521–526.

Fryling, M. (2019). Low code app development. *Journal of Computing Sciences in Colleges, 34*(6), 119.

IFR (2018). *World Robotics 2018—Industrial Robots*.

ISO (1998). *ISO 9283:1998.* https://www.iso.org/cms/render/live/en/sites/isoorg/contents/data/standard/02/22/22244.html [26 Nov. 2021].

KUKA (2019). *KR QUANTEC-2 Specification (Spez KR QUANTEC-2 V2).*

Mateo, C., Brunete, A., Gambao, E. & Hernando, M. (2014). *Hammer: An Android Based Application for End-User Industrial Robot Programming.* DOI: https://doi.org/10.1109/MESA.2014.6935597.

Morrison, J. P. (2010). *Flow-based Programming: A New Approach to Application Development*. J. P. Morrison Enterprises.

Müller, F., Deuerlein, C. & Koch, M. (2021). Cyber-physical-system for representing a robot end effector. *Procedia CIRP, 100*, 307–312. DOI: https://doi.org/10.1016/j.procir.2021.05.071.

Naumann, M., Wegener, K., Schraft, R. D. & Lachello, L. (2006). Robot Cell Integration by means of Application-P'n'P. *Proceedings of ISR 2006.*

Pan, Z., Polden, J., Larkin, N., Van Duin, S. & Norrish, J. (2012). Recent progress on programming methods for industrial robots. *Robotics and Computer-Integrated Manufacturing, 28*(2), 87–94. DOI: https://doi.org/10.1016/j.rcim.2011.08.004.

Parsons, J. T. & Stulen, F. L. (1958). *Motor controlled apparatus for positioning machine tool* (United States Patent No. US2820187A). https://patents.google.com/patent/US2820187A/en.

Resnick, M., Maloney, J., Monroy-Hernández, A., Rusk, N., Eastmond, E., Brennan, K., Millner, A., Rosenbaum, E., Silver, J., Silverman, B. & Kafai, Y. (2009). Scratch: Programming for all. *Communications of the ACM, 52*(11), 60–67. DOI: https://doi.org/10.1145/1592761.1592779.

Trower, J. & Gray, J. (2015). Blockly language creation and applications: Visual programming for media computation and bluetooth robotics control. *Proceedings of the 46th ACM Technical Symposium on Computer Science Education,* 5–5.

Weintrop, D., Afzal, A., Salac, J., Francis, P., Li, B., Shepherd, D. & Franklin, D. (2018). Evaluating CoBlox: A Comparative Study of Robotics Programming Environments for Adult Novices. *CHI '18: Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems, 366,* 1–12. DOI: https://doi.org/10.1145/3170427.3186599.

Online references

Aweida, C. (2021): Man & Machine, https://cka.co/ [9 Sep. 2021].
Bamberger Natursteinwerk (2021): https://www.bamberger-natursteinwerk.de/ [9 Sep. 2021].
Branch Technology (2021): https://branchtechnology.com/ [9 Sep. 2021].
Fab Foundation (2021): https://fabfoundation.org/getting-started/ [9 Sep. 2021].
LIT Robopsychology Lab (2021): https://www.jku.at/lit-robopsychology-lab/ueber-uns/team/martina-mara/ [12 Dec. 2021].
Quayola, D. (2021): Quayola, https://quayola.com/ [9 Sep. 2021].
Rutten, D. (2020): Grasshopper, https://www.grasshopper3d.com/ [26 Nov. 2021].
Sattlerei Niedersüß (2021): https://www.k-n.at/ [9 Sep. 2021].
Tabakfabrik Linz (2021): https://tabakfabrik-linz.at/ [12 Dec. 2021].
YOKAI Studios (2021): Robotic Fashion Design, https://www.yokai-studios.com/ [9 Sep. 2021].

Figures

Fig. 1: Boston Dynamics' Spot at the Ars Electronica Festival 2021, photo © Johannes Braumann.
Fig. 2a&b: Robotic arm controlled by an Arduino microcontroller (a, by Theremino CC), KUKA industrial robot (b), photo © Johannes Braumann.
Fig. 3: Direct manipulation of a compliant collaborative robot. Teach pendant on the right, photo © Johannes Braumann.
Fig. 4: Production immanent design for robotic fabrication, photo © Johannes Braumann.
Fig. 5: Abstracting a machine by using a robotic arm, constraining movement to geometry via flow-based programming, photo © Johannes Braumann.
Fig. 6: Flow-based robot programming with KUKA|prc, photo © Johannes Braumann.
Fig. 7: Programming an industrial pick-and-place process, photo © Johannes Braumann.
Fig. 8a&b: From manual to robotic metal polishing, photo © Johannes Braumann.
Fig. 9: Artistic interaction with a system based on industrial robots at the AEC Festival 2019 with dancer Silke Grabinger, photo © Johannes Braumann.
Fig. 10: Robots as drummers at AEC Festival, photo © Johannes Braumann.
Fig. 11: Robots playing guitar, photo © Johannes Braumann.