# Complexity and Simplicity

## *Tensions in teaching computation to large numbers of architecture students*

*A. Benjamin Spaeth[1], Theodoros Dounas[2], Joachim Kieferle[3]*
*[1]Cardiff University [2]Xi'an Jiaotong Liverpool University [3]Hochschule RheinMain*
*[1]spaetha@cardiff.ac.uk [2]theodoros.dounas@xjtlu.edu.cn*
*[3]joachim.kieferle@hs-rm.de*

*This paper describes the challenges and approaches to introduce computational thinking to a large and diverse group of architecture students during an international workshop with 300 students from different cultural backgrounds and educational levels, also integrating a diverse group of tutors whose computational expertise varied extremely. The approach suggested articulating a design task which enforced computational thinking but enabled different levels of engagement with the computer as a tool. Hypothetically this would allow all participants to engage with the computational thinking agenda regardless their computational affinity even whilst applying analogue methods. Besides the intercultural experience the workshop was successful in exposing a large group of students and tutors to the concepts of computational design whilst accommodating different learning preferences and engagement with the computer as a device.*

**Keywords:** *Computation Education, CAAD, Large Cohorts, Computational Strategies*

### INTRODUCTION

The paper presents the methods and results of a computation and architecture international workshop in collaboration with Hochschule RheinMain, Germany and Xi'an Jiaotong Liverpool University located in China. The workshop took place in Suzhou, one of the biggest and fastest developing urbanisation areas in China in last 15 years, where the question of fast and temporary urbanisation is immanent to provide living space for the thousands of migrant construction workers building up the permanent settlements. The number of students taking part in the workshop was large, numbering 300, creating an additional challenge to the delivery of computational thinking and computational exploration within the workshop for and by the students and faculty. The workshop was a one week design exercise where students from different years and levels of the two different schools from Germany and China participated in vertically integrated group.

### TEACHING COMPUTATION

Teaching computation in Architecture is challenging. It seems that everyone is expecting some failure and some success in project based teaching in Architecture, more so in CAAD (Brown 2002). CAD teaching has developed from a learned skill-based approach where the target was the emulation of the productive capacities of a drafting department of an architecture office, with few enlightened educators driving towards the design rather than the drawing fraction of CAAD. (Clayton et al 1999) Computation has been used in various supporting roles in architectural courses, from straightforward architectural design studio graphics and representation (Iwamoto 2004) to structures, construction, technology and any other technical aids. The major challenge in CAD teaching has been identified as to move from the application of a digital tool employing inefficient routines to a deeper understanding of computation and the inherit core strategies. In architecture this seems to be significantly different since architects are very likely to apply the "production bias" which detains them from getting deeper into a subject than needed for the job to be done (Senske 2014). At the same time CAAD instructors have developed a variety of strategies to assist students to understand and acquire CAAD skills and deeper understanding of computation in architecture. These strategies range from a fundamental position where CAAD is buttressing architectural composition (Flemming 1989) to reflections of a linguistic nature (Cicognani 2000, Cheng 1997); from technology being the driver (Bechthold 2007) to software seen as an environment constructed for an ease of understanding by novices (Gannon et al 2014); from CAAD as a driver for energy performance analysis (Dvorak 1988) and from integrating CAAD courses vertically in the curriculum (Bollinger 1987) to offering CAAD as a separate form of architectural design. For many CAAD courses there has been a chronic time-lag between the development of a method or tool and its introduction into the curriculum- (Clayton et al 1999). Apart from the variety of responses this last point effectively argues for the alignment of CAAD research with teaching it in the classroom, in the hope that such cross-pollination would also allow optimisation of the state of the art in skills of architectural graduates. This notion will also hopefully advance the CAAD professional practice. However there are very few studies in the literature focusing on what exactly students learn through courses in computation in architecture schools worldwide (Hemsath 2010, Senske 2014).

The educational workshop presented here is to introduce computational thinking and designing at the core of the curriculum, the design studio. The studio was organised as a vertical studio including the entire first and second year students of the hosting institution as well as all their tutors. The tutors involved in the workshop though were in the majority not computational literate which put an additional challenge to the pedagogical concept. Avoiding to be caught in an ideological clash with the 'analogue' mode of teaching which most of the participating staff was used to, the computational and parametrical thinking was introduced rather through manual and analogue means than through using computers. This challenge was approached by shaping and restricting the design exercise as to computational thinking and means are almost impossible to avoid. Computational thinking was required to set up a flexible urbanisation system from containers which could adapt to diverse situations by identifying and including recurrent parameters into the design system. The brief required students to develop a computational approach not by using computers or software but by developing a computational and parametric concept which was immanent to the design brief by its given boundary conditions and the available variables.

### *Large Group Teaching*

This workshop was conceived to introduce not only a small group of selected students with an already existing particular interest in architectural computation but to reach as many students as possible to introduce them to computational thinking in architecture and to rise their awareness of its potential, its application to design where students might have not expected it and to degrade students potential reservations against architectural computation.

Facing the large amount of students, numbering around 300, along with a diverse educational level, or lack thereof in computational tools amongst students and tutors it was a challenge to set up a robust enough design task which demanded for computation related thinking but was not only based on the use of computer or software so as also computer averse staff could tutor a student group. The workshop had tutorial groups of 20-25 students with one tutor guiding each group. In each group teams of five or four were created, each one having to develop one project responding to the brief and using computational concepts and tools. We had about 12 tutorial groups formed, with only about half being versatile in classic CAD tools and only three tutors being educated in the advanced use of computational tools. Therefore the container was conceived as to act as an invisible unit of computation, where the creation of diverse grammars and syntaxes would emerge from the experimentation with this computational unit within the set boundary conditions even without using the computer to model the systematic and algorithmic thinking.

## *The Brief*

The brief asked students to develop a small city housing 2000 people across a water way between Suzhou and the town of Zhou Zhuang, in China which locates in one of the fastest and largest Chinese urban developments at the moment. With a reference to historic urban developments along the main rivers and canals in China and by identifying the water ways as efficient means of transportations one of the determining boundary conditions was to develop the settlement along the great canal between the two permanent settlements. The second determining boundary condition was the given size of standard overseas containers and the immanent variations of stacking them. Furthermore it was determined that the containers are shipped by the floating barges only and can only be set by a specific container crane mounted on the barges. The ship, used for container distribution, was assumed that it could place a container no further or higher than 4 lengths of a container from the bank of the canal, and no more than 10 lengths across the axis. The crane was deemed exceptionally high and could place up to six containers in height. The stakeholders are people of high mobility in the Chinese urban context (migrant workers, students, early career professionals, freelancers or similar occupants). The mix and the needs of the occupants are part of the variables students had to define in setting up their parametric system. Also the different topologies which could be found alongside the canal needed to be included into the algorithm. The utilisation of a modular system consisting of containers was a means to support students' focus on computational thinking since the containers offer natural points of connection and embedded constraints. They are units that facilitate bottom up design concepts but at the same time they are architectonic through their inherently specific physical constraints and possibilities by design.

The high level strategy in developing an urban settlement should be inherently three dimensional and allow for the connection to the plots/sites of neighbouring teams. So the teams should agree on and share their specific boundary conditions of their developed concept.

## *Strategies and Computational Tools Employed*

We provided to all students introductory sessions to the computational tools and made the tools as simple to use as possible. The computational platform selected for the workshop was Rhino with Grasshopper. The faculty prepared parametric definitions on grasshopper using separately cellular automata and L-systems, which allowed the students to design using prescribed definitions. We used the rabbit plugin to be able to encapsulate high level computation on cellular automata in simple parametric decisions. The parametric definitions were well documented and sufficiently transparent so that the students would be able to employ them and understand the ramifications of their decisions in changing strategies or changing parameters. At the same time the code was accessible enough for students to inter-

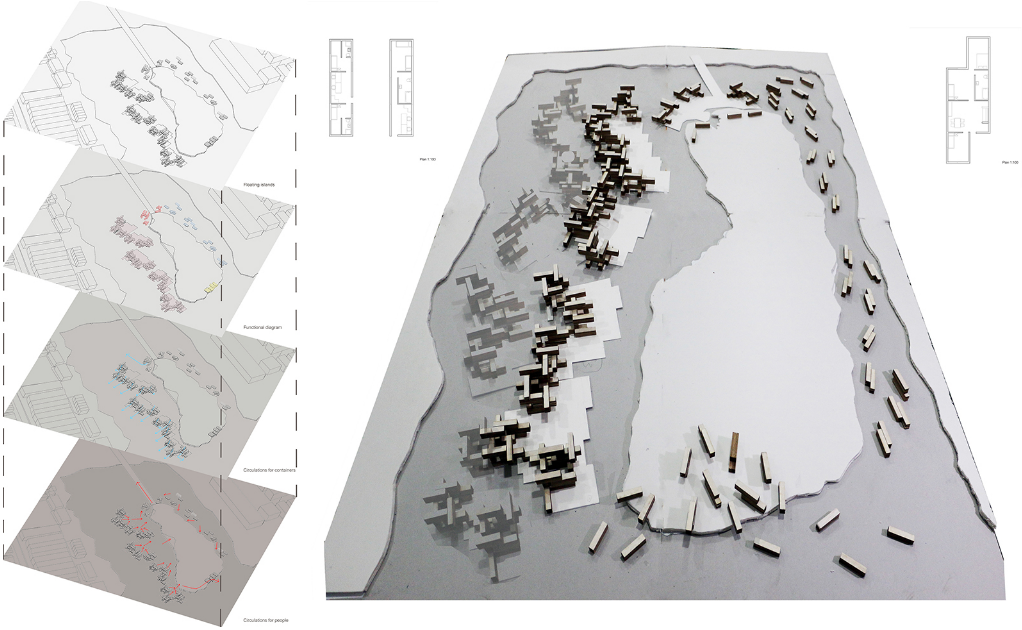



Figure 1
'Floating City'.

vene on a more fundamental level if applicable. At the beginning of the workshop an instructional session on using the parametric definitions was given to the students in groups of about 50, repeating the session 6 times. Essentially the tools were templates that the students could use. They could stay within the confines and constraints of the template or as some did tweak and explore more outside the frameworks we provided.

## RESULTS

The project 'Floating City' (Figure 1) employed not only the computational tools provided by the teaching team but the students also developed their own narrative of a floating city and a container grammar. It was noted that the grammar works on two scales, in assembling the containers into islands as autonomous entities and on a second level into assembling these islands into a city as a larger community. Students demonstrated the use of computational thinking on an internal level organising the islands as independent entities with the internal rule set applied to the elements and on an external level where they identified interdependencies between the autonomous islands and the environment articulated into a parametric system to be applied to different situations. Architectonically the work was also successful in terms of completeness and aesthetics.

The students of the 'Circular High Mobility City' (Figure 2) approached the task by introducing a twofold zoning with a commercial zone facing the land side and the residential zone facing the canal. This provided a systematic different urban quality to the different zones, introduced a public space between the zones and allowed for different patterns within the two zones to emerge. Although the project was presented as a closed ellipse the potential to be populated along the canal in different extensions is self-evident.

The project 'Floating City 2' (Figure 3) is a representative of a grid like approach. Although the computational approach would allow for overcoming a grid in favour of a flexible system of relational objects this group of students approached the task by deploying a topologically flexible grid onto the project site. Determining service points and circulation routes the grid is then populated respectively with containers.

Having started from implementing and adapting the provided L-system, combining it with the historic quadratic elementary form of the Chinese traditional city the group also ended up in a grid system (Figure 4). They conceived their grid as a rail system, where the containers can be arranged according to occupational needs. However the automated mechanism is supposed to fill the grid in the one or the other way

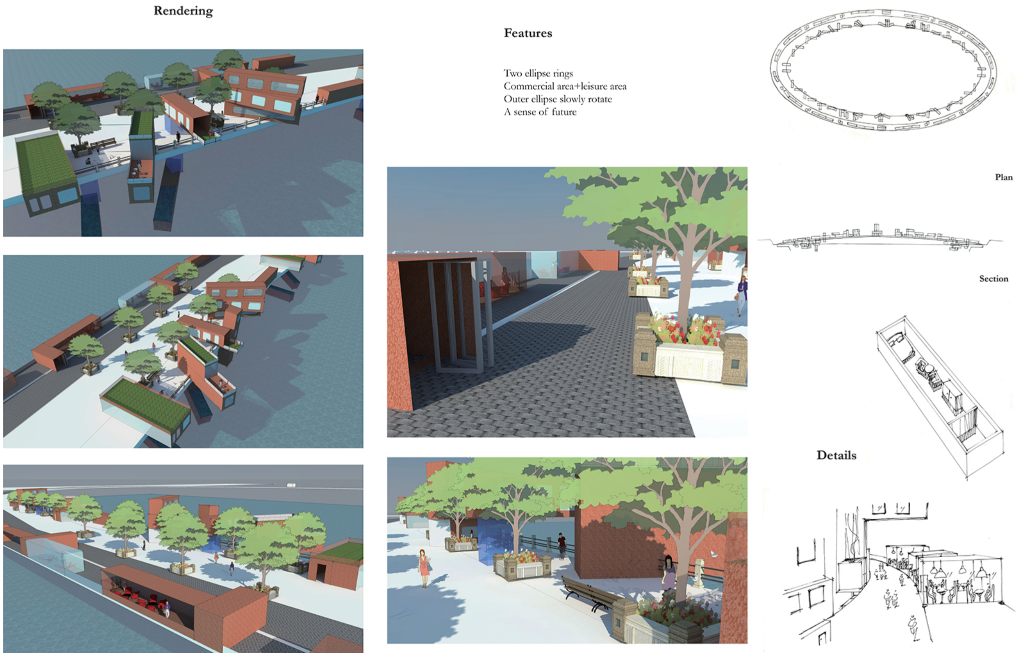



Figure 2
'Circular High Mobility City'.

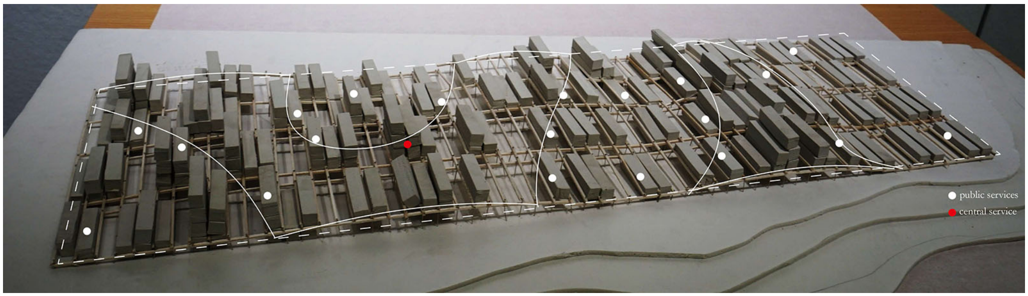



Figure 3
'Floating City 2'. Flexible grid with parameterised container deployment.

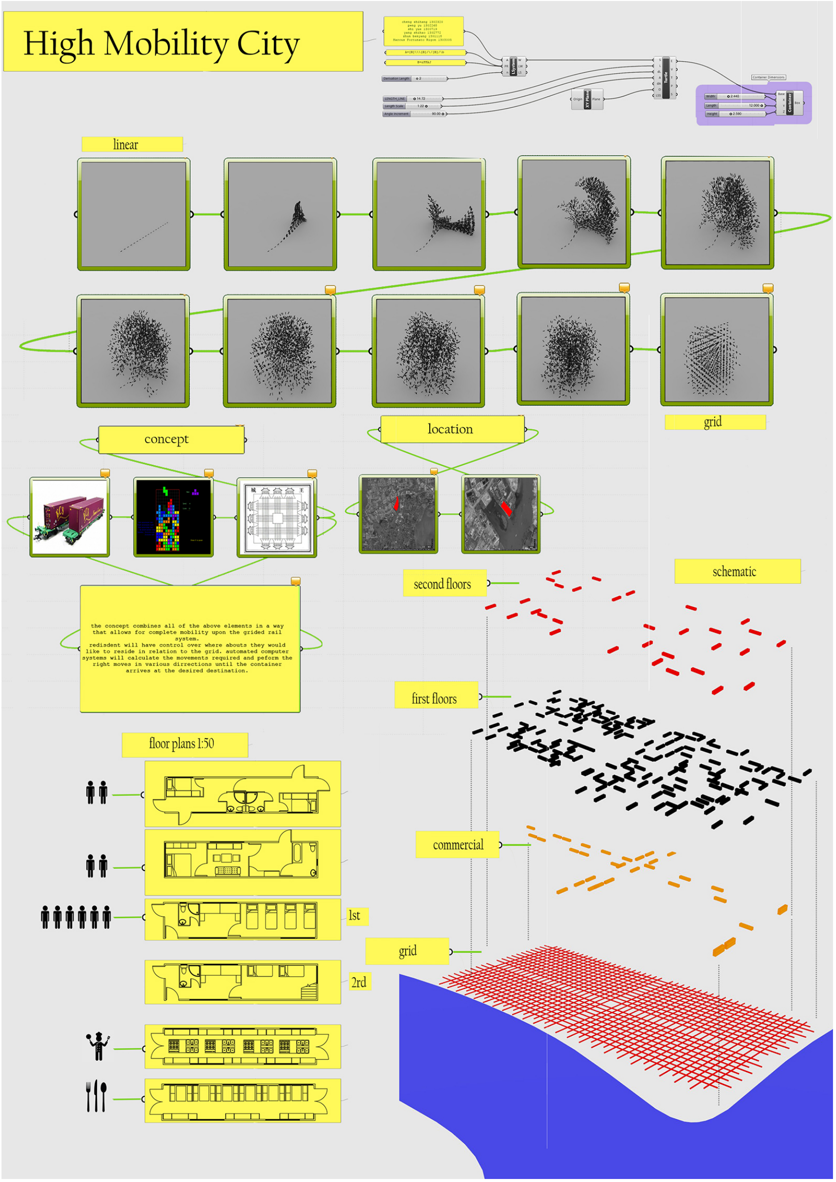
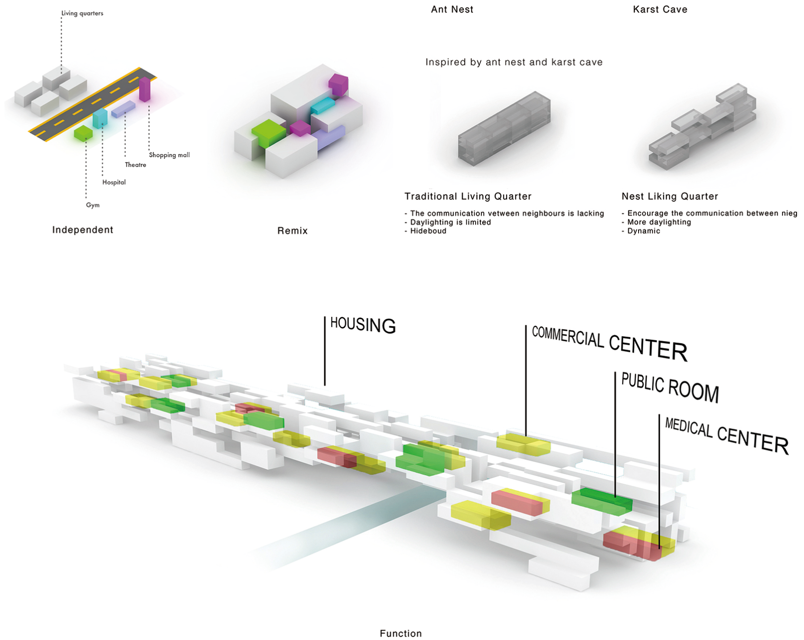


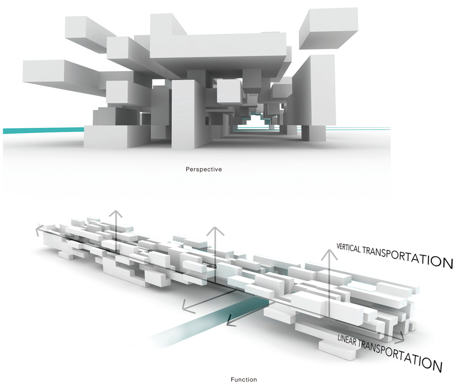

Figure 4
Rigid grid with flexible container deployment controlled by user.

Figure 5
'Nest City'. Two overlapping system patterns.




the basic principle remains the grid. This system here was conceived as an adaptable and controlled system within the rigid grid, whereas the before mentioned grid was adaptable to the found topology.

Figure 6
'Nest City'.




Inspired by the perforation of karst formations and anthills a student group developed a linear pattern which took into account daylight, topographic conditions, functional aspects as well as density considerations (Figures 5, 6). Based on a closed block definition the group loosened the block structure to aerate it but also extended it into a basically infinite ribbon that could be knitted along the canal. The repetitive pattern includes a conscious and also repetitive disruption formed by central public services. In terms of computational thinking this represents an interesting approach since the continuity is the standard case and the disruption is the exception which takes a deeper understanding since a continuous pattern needs to be overlapped with a second layer defining the disruptions.

## CONCLUSION

The projects presented in the final presentation were of a great diversity and reflected students' outstanding engagement with the task. Despite the large number of students and the assumingly neutral interest to computational methods in architecture amongst the majority of the students and tutors the engagement with the design task but also with the implicit computational agenda was on a high level. Although it was difficult to control the groups and in particular the tutors' interpretation of the design task the assignment was robust enough to engage all groups into a systematic and algorithmic approach, at least in a wider sense. Due to the container as a computational unit and the set boundary conditions most of the groups engaged with the intended computational agenda. The consciousness of the agenda amongst the students may vary but we received feedback from students that explicitly referred to the workshop and how it was inspiring for a regular housing studio in the course of their studies.

The simplicity of the interface of the computational agenda to the participating students and tutors facilitated the engagement of the large majority of participants, students as well as tutors. The level of complexity behind the computational agenda was to be uncovered by the groups themselves; on an intellectual level as well as on the level of engagement with the computational tools provided.

On a next workshop we hope that we can use the massive number of students as 'live design agents' to solve architectonic problems. Instead of programming computer agents we can use the integrated power of human and analogue computing to attempt complicated solutions to design problems. This of course can only happen if we still employ the same strategy as here: common computational templates, where the common platform is an advantage leading to diverse solutions rather than a disadvantage constraining the participants.

## REFERENCES


Bechthold, M 2007 'Teaching Technology: CAD/CAM, Parametric Design and Interactivity.', *Predicting the future*, Frankfurt/Main, pp. 767-775

Bollinger, E 1987 'The New Studio: CAD and the Workstation: Implications for Architecture Education.', *eCAADe Conference Proceedings*

Brown, A 2002 'CAAD Complements: Art and Science, Real and Virtual | Cumulative Index of Computer Aided Architectural Design', *20th eCAADe Conference Proceeding*, pp. 214-218

Cheng, NYw 1997 'Teaching CAD with Language Learning Methods', *Acadia 97*, Cincinnati, OH

Cicognani, A 2010, 'Boxing architecture', *Digital Creativity*, 11(3), pp. 190-192

Clayton, MJ 1999 'Stumbling, Backtracking, and Leapfrogging: Two Decades of Introductory Architectural Computing', *Architectural computing from Turing to 2000*, pp. 151-158

Dvorak, RW 1988 'Designing in the CAD Studio', *ACADIA Conference Proceedings*

Flemming, U 1990 'Syntactic Structures in Architecture: Teaching Composition with Computer Assistance', *The Electronic design studio*, Cambridge, Mass.

Gannon, M and Brockmeyer, E 2014 'Teaching CAD/CAM Workflows to Nascent Designers', *Proceedings of the 19th International Conference on Computer-Aided Architectural Design Research in Asia*, pp. 801-810

Hemsath, TL 2010 'Searching for Innovation Through Teaching Digital Fabrication', *Future cities*, Z\"urich, pp. 21-30

Iwamoto, L 2004, 'Translations: Fabricating Space', *Journal of Architectural Education*, 58(1), pp. 35-38

Senske, N 2014 'Digital Minds, Materials, And Ethics: Linking Computational Thinking and Digital Craft', *Proceedings of the 19th International Conference on Computer-Aided Architectural Design Research in Asia*, pp. 831-840