SPACESHIP ARCHITECTURE

A Sci-Fi Pedagogical Approach to Design Computation

PAVLOS FEREOS[1], MARIOS TSILIAKOS[2] and
CLARA JASCHKE[3]
[1,2,3] *Universität Innsbruck - Institut für experimentelle architektur Hochbau*
[1,2,3] *{Pavlos.Fereos|Marios.Tsiliakos|Clara.Jaschke}@uibk.ac.at*

Abstract. The analysis of make-belief drawings and models of Sci-Fi spaceships and architecture, leaves architects usually in absence of interior, material or program information. The spatial depth of sci-fi digital or physical models is virtually non-existent and unresolved. This discrepancy within sci-fi scenarios inspired the development of an integrated teaching methodology within design studios, with the academic objective to utilize computational methods for analysis, reproduction and eventually composition, while assessing its capacity to achieve a successful assimilation of design computation in the curriculum. The Spaceship Architecture Design Studio at University of Innsbruck's Institute for Experimental Architecture.hochbau follows a procedural approach in which the design objective is not predefined. Yet, it aims to be 'outside of this world' as a sci-fi architectural quality-enriched result of our reality, via a design oriented course with immersive computational strategies.

Keywords. Pedagogy; computation; sci-fi; academia; teaching.

1. Introduction

1.1. DESIGN COMPUTATION AND THE ARCHITECTURAL DISCOURSE

Design Computation has unquestionably established strong foundations in the con- temporary architectural profession. The main reason for this transition to a Computer Integrated Design from a Computer Aided one, is that many designers are now conscious of coding capabilities, but also the fact that CAD software advancement has enabled the extension of complex modelling capacity, the parametrization of the design space, offering flexible adaptive processes and more control over variation and design optioneering. How can design computation be taught in a way that puts it at the center of a methodology, rather than in a catalog of tools? The author's hypothesis is, that by introducing a design topic derived from science fiction, video games, comics and animation, a variety of computational design elements can be covered and explored, potentially resulting in an intrinsically computationally oriented methodology rather than a merely computationally aided one.

T. Fukuda, W. Huang, P. Janssen, K. Crolla, S. Alhadidi (eds.), *Learning, Adapting and Prototyping, Proceedings of the 23rd International Conference of the Association for Computer-Aided Architectural Design Research in Asia (CAADRIA) 2018*, Volume 1, 81-90.

Computation within elective courses typically covers just a basic understanding of algorithmic routines and digital fabrication. In contrast, the authors argue that providing students with an intangible theme, not limited to typical dwelling constraints, but rather drawing from a formal language previously established in physical models, illustrations or even text, can potentially result in an extensive vocabulary of computational means. This grammar of notions is derived from the analysis of the aforementioned scenarios both formally, and procedurally, as concepts of computer science that can generate spatial conglomerations. This accumulated database will be explored and comprehended in order to articulate a complete design proposal. The presented hypothesis is examined by an assessment of student works, while the methodologies implemented are documented and quantified.

1.2. SPACESHIP ARCHITECTURE

The bachelor design studio "Spaceship Architecture" was held for two consecutive academic years at the University of Innsbruck's institute for experimental architecture.hochbau with main objective being the fully computationally integrated impartation of design computation and fabrication skills on a level befitting a bachelor thesis in architecture, using structures and narratives from the sci-fi world (Figure 1). As Bermudez and King (2000) have argued, we are moving towards a complex reality, using Sci-Fi movies to compare different realities of a future world. Hence, analyzing and working with Sci-Fi concepts befits the contemporary academic design studio which produces increasingly elaborate and complex designs.

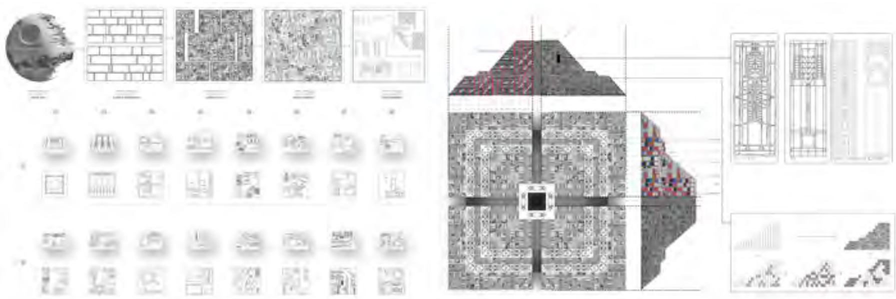


Figure 1. Case Study: Star Wars Death Star Skin Pattern Classification and Hierarchy of Elements(left) Tyrell Corporation -Blade Runner - A Sci-Fi piece of architecture as an analysis case study.(right).

Oddly enough, though, many contemporary design studios often side-line computational skills, and divert the impartation of the latter to elective courses or expect students to self-teach. As Kvan et. al (2004) have argued, though, the teaching approach to digitally oriented studios should start to shift from trying to merge computing tools with architecture to finding a new understanding of architecture through the tools. Based in part on this point, the authors argue upon an autonomous approach to design computation and investigate whether the choice of theme, as in the architectural design end goal, can lead to a paradigm shift in the way computation is being approached by schools and academics.

The most common question in complex geometrically contemporary designs is: ”how was this made?”. W.J. Mitchell explored early strategies of embedding CAD in the curriculum and noted that ”[t]he more intricate and sophisticated the solution-generation procedure, the more likely it is that we shall ask in surprise of a computer system, ’How did it ’think’ of that?”’ (1975). The Spaceship Architecture studio aims to offer a holistic introduction to complex computational methods rendering this question obsolete. The procedural thinking required for most algorithmic processes is reduced to the design context. Iteration is no more a laborious manual process, but an automated one built up from a set of initial parameters and rules associating them with each other. The studio hence deals with the question of finding the most efficient way to programmatically combining data, utilizing available tools and generating design concepts.

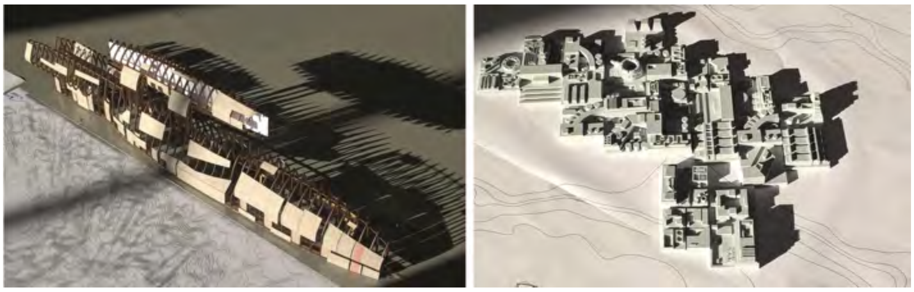


Figure 2. Physical models of the Battlestar Galactica Spaceship and Rearranged Deathstar Skin Modules as a means of understanding the complex rules behind the geometry of imaginary pieces of architecture.

To respond to the argument by Madrazo (1998), that discourses within the context of architectural design education with computers which concentrate exclusively on the tool might be misleading, the Studio provides students with an intangible theme drawing upon a formal language previously established in physical analysis models (Figure 2), in order to impart an extensive vocabulary of computational means. Preceding the original design proposal, the first semester structure of imposing a new set of restraints on a well-known sci-fi structure and asking the students to redesign or alter it, introduces them to a different design philosophy and encourages the understanding of an analysis model as going beyond an iconic generative system (Mitchell, 1975).

2. Background

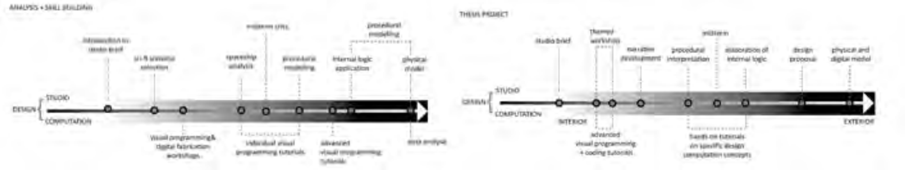
The idea of a fully computationally integrated design studio is not a new one. Indeed, it goes back to the early days of adopting computers and digital design for architectural education. Haglund and Sumption argued on the significance of a computer integrated learning. “We believe that it is important that the role of the computer be integral to the experience of learning to design, rather than acting as a catalyst to the creation of another faction. It is not desirable to have students thinking of design as one thing, technology as another, and computers as yet another." (1988). The intention of the authors for an iterative model of design through computation, is formatted around the concept where the "algorithmic" thinking is supported by traditional design methods such as model-making (Novakova et al,2010) implementing a seamless flow of data towards the digital fabrication of the model parts. The examined pedagogical paradigm can be parallelized with the learning-by-design approach of (Jabi et al. 2008) but drawing ubiquitous data not from the social or built environment, but from the hidden, though pervasive, design computation concepts within science fiction. The variety of computational concepts explored and the procedural thinking behind the designs can effectively lead to a bottom-up approach on the synthesis of form or the analysis of existing designs, with digital strategies constituting the framework from within which students elaborate their ideas, consistently pushing the boundaries of their skills.

Furthermore, the concept that CAD can and should be used to teach architectural design alongside traditional design methods is well documented in existing literature. Steinfeld argued that in order to successfully introduce CAD, it is necessary to take advantage of its potentials without imposing on the position of the traditional paper based process in designing (1988). In elaborating the structure of his experimental CAD studio he goes on to emphasize the importance of building each phase of the studio on the core idea of transferable knowledge to the next. Hence, the two-phase structure of the Spaceship Architecture studio is rooted in an array of preceding methodologies. The Spaceship Architecture studio introduces novelty to these concepts insofar that at its departure point, it detaches students from their pre-formed ideas on the conception of architectural designs by introducing sci-fi assemblies to which these notions often do not apply. Therefore, the students are forced to circumvent conceptual patterns they are already familiar with and encouraged to push their boundaries as designers. This reinforces the approach of computation as an integrated strategy rather than as one with which to enrich traditional design methods.

3. Integrated Teaching

At University of Innsbruck's faculty of architecture, digital modelling and representation skills are compulsive components of the curriculum therefore students generally are expected to be familiar with basic CAD software knowledge by the time they join a bachelor design studio. Nevertheless, it was observed over the course of the two years, that the brief and structure naturally attracted students with an above-average interest in computational design. 90% of students stated that they had used an associative modeling, or visual programming platform at least a couple of times throughout their academic path. The Studio work is structured in two distinct parts: 1) Analysis - Skill Building: digitally modeling, analysing and further elaborating a spaceship from a well-known sci-fi example in the first semester, and the formulation of a narrative. 2) Synthesis - Thesis Project: development of the same into an original architectural proposal in the second semester as displayed in Table 1.

Table 1. Timeline of the studio structure in relation to the design computation concepts taught.



At thesis level, the teaching methodology of the studio is fundamentally aimed at the production of actual architecture, therefore physical prototyping is still utilized and promoted alongside the elaboration of digital skills. There is a suggested user interface for the main implementation of computation-nevertheless, other software may be used as well as pertinent to the project in question. The workshops and tutorials throughout the year included: Nurbs modelling in Rhinoceros3D, Visual Programming with Grasshopper 3D, Visualization, Polygon Modelling, Theme-Oriented Visual Programming, Scripting with Python and CSharp within GH, Digital Fabrication and Model Making workshops. Furthermore, and as expected in a design studio environment, students participate in weekly critiques, as well as multiple pin ups, in which they are asked to present both the design narrative but equally importantly the procedural thinking behind the computational processes followed in their proposals.

4. Studio Structure

4.1. ANALYSIS

During the first semester of the course, the analysis and building skills phase, students are assigned a spaceship or sci-fi piece of spatial configuration from a pool of well-known science fiction examples, i.e. Mother1, Battlestar Galactica, Tyrell Corporation Headquarters. After understanding the brief and identifying the key elements making up the formal language of its tectonics, they begin the task of recreating the spaceship digitally, but breaking it down into simple rule defined operations. This is facilitated by a kick-off workshop at the beginning of the semester focused on building a solid base of skills and knowledge of associative modelling (Figure 3) using Grasshopper3d for Rhinoceros.

Further tutorials tailored to specific needs for the task at hand are provided throughout the semester in order to enable a toolset that allows for a seamless procedure between the visual programming platform and the CAM methods to facilitate the physical reproduction of a large-scale model. By studying the texture, structure, color, materiality of the sci-fi vessel in depth, whilst reproducing it digitally/physically, students are familiarized with a broad spectrum of approaches, tools, methods and knowledge, such as parametric and iterative modelling, shape grammars, scalar fields modelling as well as fabrication optimization, among others.

Figure 3. Parametric space representations of the cladding panel patterns of the Star Wars Millennium Falcon.

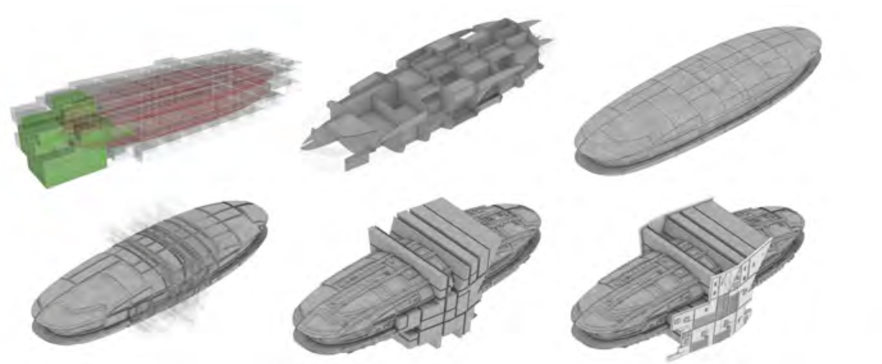


Figure 4. Translation of exterior patterns into dynamic space subdivision strategies for the interior of one of the Nebulon pods.

Building upon the findings of this sci-fi analysis serves a two-fold purpose: on one hand, it allows the students to develop a kit of computational methods and skills to draw from for the formulation of their proposal. On the other hand, it renders necessary to translate and fully comprehend their findings to then successfully synthesize them in a project. As a general rule, science fiction tectonics are mostly detailed on the outside, without a morphologically corresponding interior. Students are given the task to recognize and analyse patterns and structural logic on the outside of their chosen spaceship, and consequently to design a possible interior based on these findings (Figure 4). This “outside-inside projection” design task familiarizes them with the concept of façade-interior relations and gives them an opportunity to apply their own ideas.

At the end of this first term, the new spaceship iterations should be realized as large-scale physical models. Students learn about digital fabrication and gain confidence to actualize their digital designs. As a result, the introductory semester was organised to equip students with transferable knowledge and a sound computational foundation for their original thesis project, which is the focus of the second semester.

4.2. SYNTHESIS

4.2.1. TRANSITIONS 2.5D TECHNICAL WORKSHOP

The transition from the analysis semester to project conceptualization at the beginning of the second semester is facilitated by a second kick-off workshop. This is centered on transformational strategies, which are intended to serve as standalone design generators, while exploring the inherent generative properties of time over transcending and superimposed patterns using dynamic routines (recursion, iteration, agent based modelling). Students are expected to design and fabricate patterned 2.5D tiles using these strategies to further elaborate their skills. The workshop curriculum again utilizes Grasshopper3d as the main means of teaching computation and focuses on recursive protocols to accommodate the realization of more intricate design intentions. By exploiting coding as a design tool, students can get familiar with basic notions of computer science, such as cellular automata, L-systems, flocking boids, bin packing etc.

4.2.2. THESIS PROJECT

The thesis project brief targets a design that is rooted in a science-fiction narrative, but elaborated as actual architecture on a building scale. In the first year of the studio, the theme expanded on the development of a spaceport and on the following year, of a floating tower. The second semester reverses the "outside-inside" logic of the first. Students now build their work from ideas about interior spaces that host their story and actors, generating the outside of their design from within the framework of this structure. Both group and individual tutorials dispersed throughout the semester give equal importance to technical and design (sound narrative, architectural logic) skills. Students are expected to provide a well-structured procedural thinking, that is tightly connected to all the design computation knowledge acquired previously, but also in the way that sci-fi authors structure their stories. The design narrative is equally import to the computational/procedural logic supporting it.

Unsurprisingly, many of the projects utilized computation only to develop specific parts of the respective proposal. As is customary in the AEC industry for the most part, façade and building skin design was the main area of application for many of the proposals (Figure 5). The façade systems developed incorporated as series of methodologies ranging from simple subdivision techniques, to scalar field gradients and more dynamic arrangements employing multi-agent systems to conglomerate patterns and discrete elements within the articulation. In addition, some of the projects established a catholic computational approach for their development. In these instances, the form-finding derived purely from computational methods and characterized the whole system, including both the interior, structure and exterior. In occasions, computer graphic algorithms such as the Polygonal Scalar Fields of Paul Bourke (1994) were adapted to fit a specific shape grammar logic and form complex articulations of spatial elements (Figure 6).

Figure 5. Multi Agent trail facade articulation(left) and Scalar Field defined facade permeability (right) for spaceport designs.

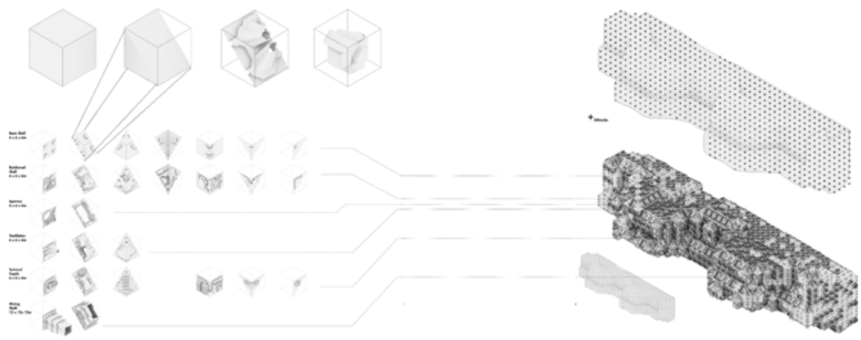


Figure 6. Polygonal Scalar Field Adaptation of Ornamental 3Dimensional elements for the Cloud Tower project.

5. Exemplary Student Project

For the first semester, the two-person team chose the spaceship Nebulon B for analysis. They found this task to be very straight forward and relayed that it helped them get accustomed to the topic of Science Fiction, and in improving their software skills. Overall, the exemplary study team found the teaching approach to be very well structured, and appreciated how the connection between interior and exterior of their spaceship and its topical relevance was highlighted as integral by instructors both in one on one tutorials and group crits.

In order to inspire the first steps of the students‘ original design proposals, a reading list pertaining to the Sci-Fi realm was handed out. It was noted that this helped them with beginning the task of formulating their own narrative, which that year was to be centered around a headquarter for a corporation operating in a dystopian future (Figure 7). They positively commented on the instructors’ emphasis on building a concise narrative with a clear connection and/or adaptability to architecture and computational design.

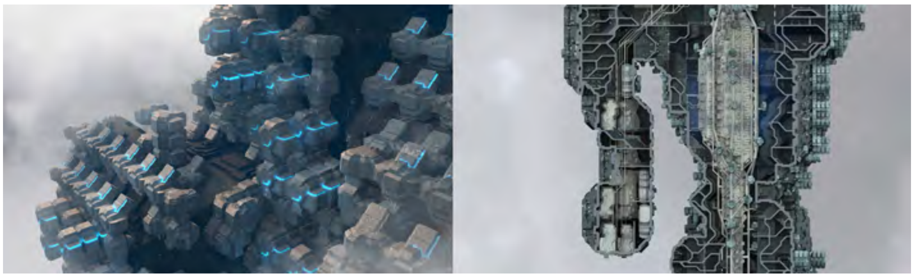
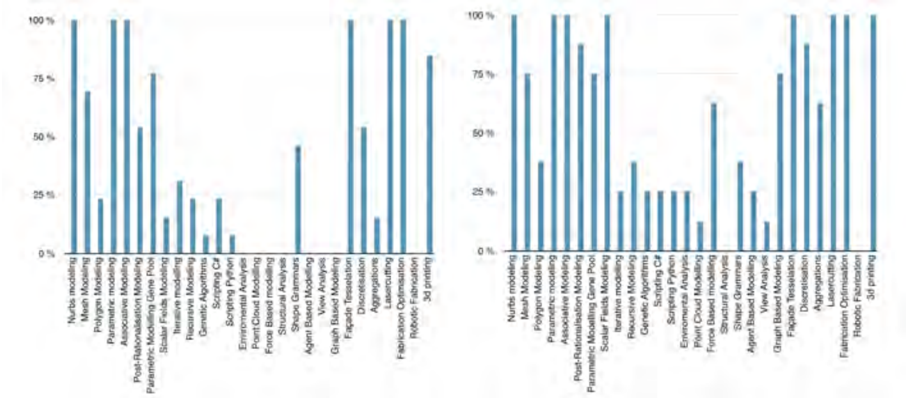


Figure 7. Perspective Visualisation(left) and Section(right) indicating the generative systems coming in play within the Nebulon team's proposal.

6. Data and Assessment

Data yielding and tracking of performance was collected through weekly crits. In addition, students were asked to answer a small survey to evaluate the timeline of the studio, and to enumerate and classify the digital tool-set that they had been provided with. In Table 2 all different modelling and design computation techniques are measured for their effectiveness and implementation as pertaining to both teaching semesters, analysis and synthesis.

Table 2. Modelling and Design Computation Techniques used in Analysis vs Synthesis.



As expected, during the second semester students managed to embrace more techniques, in contrast to the first one where Nurbs and parametric modelling were the prevailing tools. All teaching instances, such as workshops and individual tutorials, are measured for their impact on the project development for both semesters as well as both years the studio was running.Whereas the first semester shows a comparatively low amount of distribution among the projects, the second semester sees the projects diversifying in approaches and reception of tutorials. This is the result of the students gaining computational knowledge and hence themselves developing ideas and strategies regarding what software or method could benefit their proposal the most.

7. Conclusions

The main bottleneck encountered by the students was not computationally oriented; it was the translation of a shell used in movies to an actual piece of architecture accommodating a dwelling narrative. Defining clearer goals enabled the students to focus more on the explored design computation concepts and their procedural setting into a rule generated design. The increase, both in software knowledge, but also in the plethora of design computation themes explored, display a bottom-up conceptual and procedural design process which in combination with an intangible theme, results in a fully digitally integrated studio work. The learning development from simple parametric modelling notions, to iterative routines and even small scripting attempts to overcome issues arisen in a bespoke manner, displays a confident rate of success toward the achievement of a computationally amalgamated studio. Enough data has been gathered to inform further developmental steps. This data set will be used to weigh the depth to which the respective computational design strategies will be taught, depending on their evaluated project contribution rate. Regarding drawbacks encountered so far, it can be argued that while the students leave the studio with a sound knowledge of design computation, a wider range of computational approaches and specifically simulation techniques, which are most of the time undermined, could lead to further specialization and more hands-on AEC experience within a given set of students in the second semester. Future iterations of the studio may include assessments of skills in the early phases to further tailor software and teaching-level to individual students and student groups.

References

Bermudez, J. and King, K.: 2000, Media Interaction and Design Process: Establishing a Knowledge Base, *Automation in Construction 9*, **1**, 37-56.

Bourke, P.: 1994, “Polygonising a Scalar Field” . Available from <http://paulbourke.net/geometry/polygonise/>.

Haglund, B. and Sumption, B.: 1988, Toward a Computer Integrated Design Studio, *Computing in Design Education: ACADIA Conference Proceedings*, Gainesville, Florida, 291-299.

Kvan, T., Mark, E., Oxman, R. and Martens, B.: 2004, Ditching the Dinosaur: Redefining the role of digital media in Education, *International Journal of Design Computing*, **7**, -.

Madrazo, L.: 1998, Computers and Architectural Design: Going Beyond the Tool, *Digital Design Studios: Do Computers Make a Difference? ACADIA Conference Proceedings*, Cincinnati, Ohio, 44-57.

Mitchell, W.J.: 1975, The Theoretical Foundation of Computer-Aided Architectural Design, *Environment and Planning B: Urban Analytics and City Science*, **II**, 127-150.

Novakova, K., Achten, H. and Matejovska, D.: 2010, A Design Studio Pedagogy for Experiments in Collaborative Design., *FUTURE CITIES [28th eCAADe Conference Proceedings / ISBN 978-0-9541183-9-6]*, Zurich, 73-79.

Steinfeld, E.: 1988, Using CAD to Teach Architectural Design, *Computing in Design Education: ACADIA Conference Proceedings*, Gainesville, Florida, 75-86.

Wasim, J., Hall, T., Passerini, K., Borcea, C. and Quentin, J.: 2008, 008. “Exporting the Studio Model of Learning.” In Architecture in Computro [26th eCAADe Conference Proceedings / ISBN 978-0-9541183-7-2], edited by M. Myulle. Antwerpen: The Higher Institute of Architectural Sciences, Henry van de Velde. 509–516., *Architecture in Computro [26th eCAADe Conference Proceedings / ISBN 978-0-9541183-7-2]*, Antwerpen: The Higher Institute of Architectural Sciences, 509-516.