Predicting the Future from Past Experience

A Reflection on the Fundamentals of CAAD

Bob Martens[1], Alexander Koutamanis[2], André Brown[3]
[1]Vienna of Technology, Austria
[2]Delft University of Technology, The Netherlands
[3]The University of Liverpool, UK
[1] http://arch.tuwien.ac.at, [2] http://www.bk.tudelft.nl, [3] http://www.liv.ac.uk/abe
[1] b.martens@tuwien.ac.at, [2] a.koutamanis@tudelft.nl, [3] andygpb@liverpool.ac.uk

Tomorrow's architectural practitioners seemingly need to gain an overview of, if not master, a wide range of computer aided architectural design applications, from image making to Building Information Modelling (BIM) to digital fabrication. However, we are sceptical whether there is wide recognition that there is value in a broader appreciation of the underlying principles that organize these applications. CAAD software, once an exploration of architectural ideas, has become a commodity. But as digital tools have become more ubiquitous the relationship between practice and research has, broadly speaking, become more ambivalent. What has been lost, and what gained, in this change?

Keywords: *Fundamentals; computational design; scenario; architectural practice.*

Introduction

We sketch here an argument, accepted by most of those working in CAAD research, but not necessarily by those involved in architecture more generally, that computer-aided architectural design involves something more fundamental than is apparent in the suite of software tools in which the profession and the schools are awash. The construction of software to support design implies, indeed demands, a deep understanding of the processes by which design and designers work. Yet this understanding is at best implicit and arguably absent from the tools that pervade the marketplace. In the spirit of Maver's *Seven Deadly Sins of CAAD* (Maver, 1995) we maintain that a close examination of the fundamentals of computational design is more important than ever, despite, even because, of the broad adoption of computer-aided tools. This has become even more urgent because the scope of design and building automation has increased and this has direct influences on areas that were originally in the periphery of CAAD (e.g. robotics in building construction or ubiquitous computing). Understanding how the fundamentals of CAAD operate in these extended areas as well as anticipating emerging applications is important for the effective development of architectural practice in the digital era.

However, what we have witnessed is a relationship between CAAD in the academic world and computerization in practice that can best be described as ambivalent. We can certainly claim that CAAD has paved the way to computerization by developing computer systems, adapting architectural practices to computerization and by exposing students to computing applications and the underlying approaches to architectural design. On the other hand, it can be said that the influence of CAAD research has been less widespread and less influential than might be expected, given the growing ubiquity of the computer and digital applications in the last decade of the 20th century. The systems used in practice arguably owe more to commercial development (generally based on transfer from other design areas) than to CAAD research and development and the consequent products and approaches.

This ambivalence derives, to a large degree, from the two complementary foundations of CAAD; theory and technology. CAAD theory stemmed primarily from the rationalist line that it could be said typified late modernism. Its outlook was primarily introspective: despite the many external influences and references, its sources and targets remained firmly conventionally architectural. In later years the external sources became more pronounced, especially from areas such as artificial intelligence and cognitive science, which were characterized by similar preoccupations with human and machine intelligence. Nevertheless, theory retained its coherence over the years as well as its emphasis on automated design (Steadman, 1976, 1983; Stiny, 1975, 1980).

The technological component has been stable primarily with respect to application areas such as drawing, modelling and visual presentation, where for quite some time computerization competed with analog practices. CAAD's reliance on technology transfer from computer science and related disciplines meant that sudden changes in the implementation means and computing environments were not uncommon. Moreover, such changes also influenced CAAD theory and methodology (witness the importance of the Internet for the development of ideas on collaboration and communication), sometimes leading to bandwagon jumping.

The popularization of the computer has made the technological component the apparent priority in CAAD education. Tomorrow's architectural practitioners need to gain an overview of, if not master, a wide range of applications, from image-making to building information modelling to digital fabrication, with emphasis on acquiring skills with specific computer systems. It is less clear to practitioners that they need to understand the underlying principles that organize these applications. CAAD software, once an exploration of architectural ideas, has become a commodity. This influences not only the attitude of practitioners towards computer use but also restricts the scope of their interests to available rather than relevant applications. For example, design analysis and evaluation, once one of the prominent areas in CAAD, is currently available only in superficial normative code-compliance checks if at all (Maver, 1978). The availability of affordable and efficient simulation software and the widespread interest in building performance (especially with respect to energy) are apparently not enough to stimulate demand in this area.

The history of CAAD is short but rich in rapid developments (Bhatt, 2006). The character of the field has changed little in the course: ambitions, goals and means remain largely unchanged since the early years. What has been changing is the position of the area relative to architecture and building in general, both in academia and in practice. This has led to changes in the internal priorities of CAAD, especially in teaching. A critical examination of the strengths and weaknesses of the area leads back to the fundamentals of computational design. These are more important than ever, despite, even because, of the broad adoption of computer-aided tools because they determine not only the true character of the area but also possible scenarios for new directions for CAAD research and development.

Fundamentals of CAAD

Let us first consider some challenging questions related to the realisation of widely known archetypical building forms. When Eskimo's erect an igloo, are they in need of a BIM (Building Information Model)? Or, when the Dogon in Mali setup an adobe dwelling, do they need a drawing? Similarly, the Pueblos in New Mexico: Are they in need of precisely prepared building data? The answer is for sure "no". Without doubt these examples could be replaced by other ones, e.g. similar types of housing which do not require extensive written documentation because they do not require an official building permit and hence are not subject to product and process regulations devised by a professional or government body. The creation of these buildings follows a pattern which is in line with construction traditions as they are handed down through the generations and, ideally, is subject to case-by-case optimization. The erection of the building is uninterrupted, i.e. the *designer* and the *executing architect* are either one and the same individual or are at least both continuously involved. While the repertoire of patterns is limited, slavish adherence to the basic patterns is not the central task in this exercise (Rudofsky, 1965). Furthermore, any accumulated experience is directly transmitted orally while the practical aspects are learned by subsequent generations by simply doing (tutorial situation on site).

The stability and continuity of traditional building design and construction in the face of modern social and technological developments can be interpreted as indifference to such developments (Hall, 1990). What is then the case with a modern building? Sydney Opera House is particularly interesting in that it lies in an era that defines the cusp of computerization in the design process (Fromonot, 1998; Brown, 2001). Its form was redefined from Utzon's original concept to make it possible to build and analyse using the computer, by rationalizing the geometry. Peter Rice was included on the team at Sydney because of his computational skills and understanding of the links between the geometry, the engineering and the digital representation. These skills allowed him to write a computer programme, whilst on site, that solved the major problem that was preventing construction from continuing, of how to set out the doubly curved form in 3D space. What Rice was working with was an array of numbers as output from the computer. Only because of his intimate knowledge of the software was he able to interpret those numbers; it seems, by being able to turn them into a picture in his head.

Often we forget in architecture about the debate on whether computers should be analogue or digital. Because computers became a digital device they suffered (and still do suffer) from the fact that they are good at number but not at shape. Steve Coons, writing in *Soft Architecture* Machines in 1975 reflected on the Computer Graphics issues that were constraining the development and application of the computer in architecture (Negroponte, 1975). He noted, for instance, that 'very little work has focused on the graphical abstractions and nebulous interactions commonly found in human discourse accompanied by computer graphics'. It has been the development of graphic interfaces and graphic capability that has placed the computer in the position that it occupies in architecture today. But that has generally been via massive increases in computer power rather than fundamental rethinks of how graphic capabilities can be extended and refined. And this year Microsoft dropped OpenGL just to add another shackle.

However, if we accept CAAD as an integral part of the new framework of architecture and building, what are the fundamentals of CAAD? How can we justify the need of computational design in architectural processes? The first thing we notice is that design computing has already entered most information streams in practice. The production of information in architectural practice has long taken place by means of drawing documentation. Planning can be regarded as "looking ahead" and leads to the creation and maintenance of documents which depict a projected state. As the repertoire of building designs increases and goes beyond a few continuously reproduced types, building actually results in a one-of-a-kind product. If the *designer* is not at the same time the *executing architect* it is essential to provide planning details. There are three aspects which require particular precision in planning details:

1. more people involved;
2. higher degree of task sharing;
3. indirect communication of information.

The graphic visualization of the intended building is primarily a visual aid to understanding the design. More important in many respects are the numerical data that come with the visualization. They represent the authoritative part of the design.

What did CAAD initially change in this respect? Not that much at a first glance, besides the computerization of working habits, in other words a reproduction of existing procedures. The drawing itself is no longer purely analogue but still behaves as if it were analogue. Even the so-called innovations of computerization have a clear analogue origin, e.g. layering which derives from analogue overlay drafting. Design computerization has extended the concept but primarily in a quantitative sense: in the early days of CAAD the number of layers a software package afforded was a main performance criterion.

Direct transfer of drawing to the computer did little to improve the structure of designing. Procedures still suffer from a fragmented way of working, as bits and pieces are being stuck together and interpreted as a whole by the user. In other words, design documentation remains based on personal interpretations of a conglomeration of unlinked drawing parts that may (and will) potentially contain inconsistencies and contradictions.

Which were the typical promises CAD software distributors made in the past? The very first argument to be mentioned is increased productivity. Computational design was supposed to facilitate daily office work and foster an effective if not congenial link between man and machine. 'User friendliness', which used to be the most effective sales point, is no longer used as a killer argument. The broad availability of computer systems has led to a decrease in the number of users without any prior computer experience.

What can CAAD do? While the blueprint is still a strictly two-dimensional representation of numerical data, digital design allows considerably enhanced access to the data contained in the design. In this case not only the numerical indications count, but also the data on which the graphical visualisation is based. A computer-supported plan may be used to extract measurements, provided that an adequate method of data exchange is used. Metaphorically speaking, the blueprint develops a third dimension. The individual information levels can be used in isolation, data and all associated information can be extracted from and entered into the design.

The previous distinction between architect and draftsman has long been abandoned; architects often not only create computer files, they also maintain them and enter any update to a design directly into a file. Parallel to the development of CAD packages, visualization software emerged and gained an increasing importance in architectural practice. Some practitioners also became at image processing, touching tricky renderings and creating impressions of lightness and transparency.

Where is CAAD going? If one develops the concept of data input and output in the course of planning, the next step towards a *common knowledge base* (e.g. building information models) seems logical. While the journey towards this aim is fraught with obstacles at present, the vision can already be discerned on the horizon. It is only a question of how long it will take until the necessary tools are ready for mass use. What ultimate form they will take, and how they will gradually be improved to fit the requirements of practice remains to be seen. At any rate these tools have a number of tasks to perform, and an uncomplicated user-interface is certainly a priority. One of the core tasks will be the administration, handling and management of the growing amount of (modifiable) information that is involved in a building: the sharing of recorded knowledge, accumulated experience on commonly available resources with open access to stored expertise at any time.

Scenario thinking

The complexity and extent of current design automation make any attempt to reinforce coherence and comprehension in CAAD by reference to its theoretical basis a cumbersome and delicate exercise. In order to reduce it into a manageable proposition we can apply scenario techniques that facilitate anticipatory thinking (Godet, 2006; van der Heijden, 1996; Schwartz, 1991). Scenarios combine facts with possible trends into models of alternative development paths. Forecasting the impact and risk of each scenario is useful but frequently inadequate, as modelling may fail to include important factors. Probably more useful for the analysis and evaluation of the current situation and future states is the mapping of variability within each scenario as well as possible relationships between scenarios. The resulting network of states and key factors or developments in all scenarios forms a primary source of decision taking as well as a test bed for the effects of decisions.

Most scenarios for the revitalization of the fundamentals of computational design depart from current CAAD curricula. These express not only educational priorities and capabilities but also indicate research productivity (rather than effort). The two main features of current curricula are the aforementioned overview of a wide spectrum of CAAD applications and integration. The latter takes two alternative forms:

1. Inclusion of general architectural knowledge and tasks into CAAD courses so as to provide more than training with technologies. This is a traditional CAAD attitude and has often led to strained relationships as CAAD tended to supplant other specializations by stressing the modernity and efficiency of computational tools against conventional domain knowledge and practices.
2. Inclusion of CAAD elements in courses belonging to other aspects and specializations. This is a relatively recent tendency and reflects the effects of the democratization of the computer, i.e. an increase of architectural interest in the computer and a parallel weakening of the position of CAAD as custodians of computing technology in architecture.

The combination of technological width and the two integration forms results into a higher degree of fragmentation in CAAD curricula than in other specializations. This promotes on the one hand the dissemination of CAAD knowledge but on the other weakens the coherence and cumulative effect of CAAD courses. As a result, the main stakeholders in any scenario are CAAD educators, with their academic institution a possible ally in attempts at curriculum improvement.

While the identification of such basic trends can be straightforward, the driving forces in the development of CAAD can be complex and obscure. Probably the most important factor in that is the ambivalent relationship between CAAD and practice (including the developers of software for practice). CAAD is rather eclectic in assuming a leading role in practical design and construction automation. Some applications such as digital fabrication have been initiated in or have at least benefited from CAAD research and education, while others (including drafting and building information modelling) are considered either at a very practical or a very theoretical level.

These conditions suggest that the key uncertainties in all scenarios concern the relationships between (a) CAAD and other specializations in architecture and building, and between (b) CAAD and software producers. These uncertainties also relate to the possible extreme future states of the area:

1. *Dissolution*: CAAD is a temporary area that has stimulated design computing but has ultimately to be dissolved and its knowledge distributed to

other specializations in architecture and building
2. *Theory*: CAAD is a theoretical area that propagates a particular approach to design; attention to the technological component is a necessary evil that has been partly alleviated by the democratization of the computer
3. *Support*: CAAD will become the technical support of designing with computers and play second fiddle to other specializations and areas.

Evolutionary scenarios

Evolutionary scenarios represent a logical continuation of the existing situation in CAAD. They accept as initial state the current distribution of subjects among several courses and the dependence of these courses on extrinsic factors: commercial software development, priorities in practice and the utility of computational tools to other specializations. In this state the role of CAAD varies from connecting tissue to technical support. Evolutionary scenarios stress and enrich this role to the benefit of coherence, comprehensiveness, consistency and utility. Consequently, they have two interrelated goals, one for CAAD and a second for architecture in general.

The goal for CAAD is to regain a strong identity, similarly to the 1980s and early 1990s, by presenting a coherent theory and consistent techniques and tools. This would reduce the appearance of arbitrariness and opportunism in current CAAD education and provide connections with permanent or topical problems and preoccupations in architectural design and construction. The presence of CAAD as a theoretically strong, integral area is a prerequisite to achieving this goal.

The second goal refers to the utility of CAAD knowledge to the wider development of architecture and building in the electronic era. The socio-technological changes of this era have already started transforming the design, construction and management of the built environment but admittedly with few economic or performance-related benefits yet. CAAD has the potential to become a major driving force in this transformation, by interpreting and applying general trends to architecture and building. An alternative (assuming that other specializations are capable of taking over parts of CAAD technology and knowledge) are de-central models where smaller, specialized CAAD cores stimulate the development and application of computational systems in different aspects. In both situations the coherence of the theoretical framework of CAAD is essential effective and consistent communication.

Between these two extreme situations there is a third condition, where CAAD achieves coherence partly by reducing its application spectrum (an ongoing development with advanced technologies such as simulation) and consolidating its activities to either design information processing or design automation (generation). The main advantage of this is the attenuation of technological problems (as CAAD focuses on a relatively compact corpus of technologies and applications) to the benefit of a strong view of the relationship between computerization and architecture, resulting into a product that forms the basis for most applications (including those delegated to other specializations).

The main uncertainty in evolutionary scenarios refers to the parallel mode of development in practice and academia. As practice is primarily served by commercial research and development, academic research and teaching essentially follow practice by providing students with relevant skills and a deeper understanding of what and how these skills serve. The influence of CAAD is expected to grow as effectiveness and reliability improves on the basis of CAAD knowledge which guides use of the technological tools. As a result, CAAD remains dependent on commercial developments and their acceptance in practice.

Evolutionary scenarios are characterized by the correlation of such typically bottom-up developments with the top-down theoretical component of CAAD. Mismatches between the two identify either missing tools or theoretical lacunae and inconsistencies. Such conflicts may lead to focused development but, given the width of the CAAD spectrum, may also lower the priority of such development and lessen interest in related problems. In either case, the sensitivity of CAAD to commercial research and development increases. On the long term it is conceivable that all necessary tools will be made available. However, the current state of the art shows a patchy picture, with some application areas heavily saturated and others, arguably equally important and lucrative, rather neglected. Consequently, the top-down theory of CAAD is constrained by an arbitrary bottom-up framework of applicability and usability.

Revolutionary scenarios

Revolutionary scenarios depart from the low expectations one may have from the current situation in CAAD and by extension from the expected low impact of evolutionary scenarios in a time frame of 5-10 years. Rather than relying on external resources and priorities, revolutionary scenarios return to the basics of CAAD and concentrate on the development of fundamental solutions, ranging from applications not yet available in practice to alternatives to existing commercial systems. Of paramount importance in these scenarios are products that cross over from the realm of CAAD theories to practice, as well as research results that form the basis of educational activities. In order to do so CAAD research has to establish strong theoretical and methodological foundations for specific problems and invest time and effort into producing working prototypes (as opposed to the more familiar demonstrations).

Judging from CAAD research output in the last decade there are sufficient products that challenge, augment and enrich existing knowledge and tools. However, current funding frameworks provide few incentives for the further development of these products into systems usable in practice. A common solution is to form alliances with commercial parties, which may lead to promising ideas disappearing in corporate chaos or drowning in compromises and extrinsic constraints (ironically things they may have set out to challenge and change). Often the highest expectation is to establish a focused and well-funded research group with more influence on academia than on practice.

Revolutionary scenarios rely on such groups which develop further into driving forces for specific, usually compact sub-areas. The resulting picture of CAAD is one of a collection of islands, each with an own specialization and loose connections with the others. This permits each island to form an own identity, partially by means of conflict with existing tendencies as well as by competing with each other. Productivity can be high even if restricted to a small application area, partially thanks to technological opportunism. Moreover, the products of each island can have a higher vertical consistency and completeness due to the necessity to develop research results at all levels: theoretical, methodical, algorithmic, implementation (Marr, 1982).

The impact of such islands and their products is initially restricted to the academic world (advanced studies and research). Influences on practice are indirect, through the work of educators who train there, are influenced by their (theoretical) products or use their prototypes in teaching and research. The bottom-up framework in revolutionary scenarios inevitably results into a pluriformity of possibly deviating or even conflicting ideas. This may seem an obstacle to the development of a single, all-encompassing theory of computational design. However, the value of developed and tested ideas about specific issues should not be underestimated. The depth and strength of local inquiries should provide sufficient prowess and common elements between them to justify CAAD. Moreover, the continuation of the area in these scenarios depends more on local research results than on a general theoretical framework.

The key uncertainties of revolutionary scenarios refer firstly to the competition with commercial products and secondly to the problem of stimulating long-term research and local cooperation in the face of growing academic bureaucratization. The resources of academic research are significantly inferior to those of commercial enterprises which also possess established support structures for the labour-intensive last stages of development, as well as for product distribution and user support.

Quantifying effects and making decisions

The usual last stage in scenario planning is the quantification of the effects of each scenario so as to arrive at decisions supported at least by comparative evaluation. This would be possible for a particular school with the proviso that the uncertainties in each scenario are substantial. Doing it for the whole area of CAAD is obviously futile. Far more interesting is the comparison of the two scenario classes, especially with respect to common premises and conclusions.

Evolutionary scenarios suggest development at a quiet, steady and uniform pace, while revolutionary ones propose that revitalization of CAAD can rely on local, possibly explosive growth. In the evolutionary case change is wide and controlled (primarily by a theoretical corpus that is not far removed from the one established in the 1970s). Revolutionary scenarios are motivated by the same theory but at the same time keen to challenge and reform it.

Both scenario classes illustrate the need to reinvest in the theoretical component of CAAD but also make evident the importance of the technological component either as proof of the capacity of the theory to improve architecture and building or as a foundation of hypotheses, choices and methods. The main difference lies in the selectivity of revolutionary scenarios which cannot rely on commercial products and therefore have to narrow their focus and scope.

The principal uncertainty in all scenarios concerns the acceptance of the contribution of CAAD by practice and other specializations. In the evolutionary ones CAAD must be accepted as the agent of computational technology transfer to architecture and building, a role that could be dismissed as superfluous both in practice and academia if the added value of CAAD cannot be demonstrated beyond the level of technical support. In revolutionary scenarios the importance and relevance of the alternatives proposed by CAAD must be made evident by their performance (primarily effectiveness and reliability). To achieve such acceptance both classes rely on the coherence and elucidation provided by the theoretical component.

Some conclusions

It is worth reflecting here, on two salutary papers written ten years apart. According to Burry (2005), in his thoughtful evaluation of the contemporary disjunctions that are evident, and referred to above, are only going to be tackled, 'when CAAD research is undertaken conjointly within teaching and practice can the links be properly formed between the two'.

Maver (1995) noted that 'It is extraordinary to observe, with increasing frequency, the emergence of 'new' ideas in the field which have striking similarities to early, abandoned and almost forgotten work …'. In this context we should note that firstly, this is still true. Secondly, we may be guilty. Similar reflections to the reflections that we have presented here, have been made by others.

This paper is put forward as a position for debate. It is the result of a collaboration and open discussion between the authors. We do not all agree; we have different views. This paper is in some ways a compromise, in some ways a case of one person's view that the others accept as valid. And this is the point that we aim to make. CAAD research should not descend into a situation that simply takes current tools and technologies and sees what can be done with them. If debate about aspects relating to the philosophical, the cultural, the educational, or suchlike, is lost, then the field becomes devalued.

References

Bhatt A.; Martens, B.: 2006, ON-TO-CAAD: Investigating the Knowledge Contained within the Corpus of CAAD Research, CAADRIA 2006 [Proceedings of the 11th International Conference on Computer Aided Architectural Design Research in Asia] Kumamoto (Japan) 2006, pp. 413-424.

Brown, A.: 2001, Peter Rice: The Engineers Contribution to Contemporary Architecture, RIBA/Thomas Telford Publications, 2001.

Burry, M.: 2005, Digitally Sponsored Convergence of Design Education, Research and Practice, Computer Aided Architectural Design Futures 2005 [Proceedings of the 11th International Conference on Computer Aided Architectural Design Futures, Vienna (Austria) 20–22 June 2005, pp. 3-22.

Coons, S.: 1975, Computer Graphics, in Soft Architecture Machines, MIT Press, p. 59.

Fromonot, F.: 1998, Jørn Utzon: The Sydney Opera House, Milan, Electa.

Godet, M.: 2006, Creating Futures: Scenario Planning as a Strategic Management Tool, Economica, Paris.

Hall, E.T.: 1990, The Hidden Dimension, Bantam Doubleday Dell Publishing Group.

Heijden, K. van der: 1996, Scenarios - The Art of Strategic Conversation, Wiley, New York.

Marr, D.: 1982, Computer vision, W.H. Freeman, San Francisco.

Maver, T.: 1987, Software tools for the technical evaluation of design alternatives, in T. Maver and H. Wagter (eds), CAAD Futures '87. Proceedings of the Second International Conference on Computer-aided Architectural Design Futures, Elsevier, Amsterdam.

Maver, T.W.: 1995, CAAD's Seven Deadly Sins, Sixth International Conference on Computer-Aided Architectural Design Futures, Singapore, 24-26 September 1995, pp. 21-22.

Negroponte, N.:1975, Soft Architecture Machines, The MIT Press.

Rudofsky, B.: 1965, Architecture Without Architects: A Short Introduction to Non-Pedigreed Architecture, University of New Mexico Press [reprint], Albuquerque.

Steadman, J. P.: 1976, Graph-theoretic representation of architectural arrangement, in L. J. March (ed), The architecture of form, Cambridge University Press, Cambridge.

Steadman, J. P.: 1983, Architectural Morphology, Pion, London.

Stiny, G.: 1975, Pictorial and formal aspects of shape and shape grammars, Birkhäuser, Basel.

Stiny, G.: 1980, Introduction to shape and shape grammars, Environment and planning B, 7, pp. 343-351.

Swartz, P.: 1991, The Art of the Long View Planning for the Future in an Uncertain World, Doubleday, New York.