CAAD’s Seven Arguable Virtues

Alexander Koutamanis
CAAD’s Seven Arguable Virtues

Alexander Koutamanis

In 1995 Maver proposed seven deadly sins for CAAD as the prelude to a critical discussion on the principles, methods and practices of the field. In an attempt to return to this discussion, in this paper these sins are linked to the seven deadly sins and the complementary seven heavenly virtues. The analysis of computational processes into a theoretic, an algorithmic and an implementational framework provides a framework for the positioning of sins and possible or already attained virtues in CAAD.
1. A tale of two iconoclastic papers

The opening of the 2003 eCAADe conference at Graz was dominated by a short homage to Tom Maver who had reached retirement age earlier that year. Among the many tributes and jokes there were a couple of facts that intrigued me. It appears that Maver’s most cited paper is his contribution to the 1987 CAAD Futures conference, entitled “Software tools for the technical evaluation of design alternatives” (henceforth “the 1987 paper”) [1]. In second place came his probably widest known paper, from the 1995 CAAD Futures conference, “CAAD’s seven deadly sins” (henceforth “the 1995 paper”) [2]. I was initially surprised by the reversal. One would expect that the subject and character of the 1995 paper would have made it a clear citation favourite. Also in terms of timing it made little sense. Publications from the mid-1990s, when CAAD was peaking as a scientific and technological field, tend to dominate the literature and reference lists in current papers, articles and books, especially if the authors are among those we consider to be the founding fathers of CAAD.

The initial surprise was soon followed by the realization that the 1987 paper is not only my personal favourite among Tom Maver’s output but also a standard reference in my own work. Despite its age the paper is still an accurate description of problems we are facing in the computational analysis of architectural designs. When I first read it in the late 1980s, the paper proved a true eye-opener. It was written by a leading authority in the field and come from a research group that had then the most experience with the subject, ranging from the abstract levels of theory to the practical levels of software production and dissemination. Nevertheless, it contained no trace of complacency. Instead it made explicit limitations of existing approaches and outlined challenging new directions for research and development. My belief in the possibilities of automated design analysis and my interest in simulation, complexity and the management of complexity were stimulated, supported and guided by ideas expressed in the 1987 paper.

By comparison the 1995 paper had a lesser influence on me. This time I had the pleasure of enjoying it live as a keynote paper at the CAAD Futures conference in Singapore. I clearly remember the joy with which the conference delegates welcomed Tom Maver’s provocative statements. Strangely enough many of us were in agreement with his arguments and warnings. CAAD was reaching maturity. The pace and character of technological development seemed to ensure a stable future for CAAD and a great influence on architecture and building. Things could not have looked brighter. Still, this made the cautious and critical members of the CAAD community even more alert to the dangers of over-optimism. Maver’s 1995 paper managed to voice some fundamental concerns, in particular the need for higher coherence, consistency and performance.
Superficially the 1995 paper can be seen as a logical successor to the 1987 one. If one substitutes analysis with CAAD in general, add a snappy metaphor and focus on the conclusions, one can arguably transform the 1987 paper into the 1995 one. However, there are essential differences between the two papers. The 1987 one seems to conclude a stage in the development of an area with a critical evaluation of actual achievements. This evaluation is remarkable in that it pushes the user and application requirements far beyond what would have been considered sufficient at that time. Even today architectural practice still seems largely satisfied with the limited techniques criticized in the 1987 paper. The few problems that cannot be handled by such techniques are delegated to specialists, usually only remotely linked to the design process. The 1987 paper proposes a completely different development direction towards empowering the designer with advanced tools that provide precise, accurate analyses and design guidance rather than summary justification. In the 1995 paper new directions are only implicit, apparently in an attempt to open a wider discussion on the fundamentals of CAAD. This might not be merely a matter of text length (the 1995 paper covers just two pages) or purpose (keynote). A very good reason for doing so is the enormous width and breadth of the subject itself: the theory, methods and products of CAAD. Rather than promoting a specific approach the paper is limited to the identification of weaknesses and errors, which in many instances should be rectified simply by doing things properly.

The common element between the two papers is their iconoclastic tone. With little sensitivity or respect for established ideas the papers propose radical reforms and urge the reader to embrace these reforms in order to avoid imminent, grave dangers. This is particularly evident in the evaluation of precedent research which ultimately amounts to a wide critical rejection. Rather than politely highlighting positive aspects of what has already been achieved and proposing incremental improvements and augmentations, Maver attacks fundamental beliefs and common practices in a manner that leads to extensive changes at all levels. In both papers he challenges the field of CAAD to destroy existing false icons not out of revolutionary zeal but in order to improve or revitalize itself on the basis of rational, constructive reasoning.

In the intervening years the challenges of the 1987 paper have been present among the themes of CAAD research. Admittedly synthesis and representation remain the clear favourites in mainstream CAAD but analysis attracts sufficient attention for the transfer of advanced technologies and the precise and accurate exploration of building behaviour and performance. Even though the results of such research have yet to become standard instruments of design guidance, computational and automated analysis of building designs is arguably adequate for the evaluation of a large number of aspects. Unfortunately a number of key
issues remain unsolved, especially concerning the relationship between analysis and synthesis. However, these are mostly general problems of design computing rather than issues particular to analysis.

On the other hand, the discussion the 1995 paper should have initiated has yet to take place in a satisfactory way. As CAAD was reaching maturity and such a discussion seemed inevitable, the democratization of the computer (also known as the electronic revolution) overtook it with a rapid and increasing pace. CAAD lost its role as scientific and technological elite with almost exclusive rights to expensive, rare computing facilities. Other academic specializations too gained direct access to affordable, powerful computer applications for their own areas. Moreover, architectural practice finally embraced computerization not because of the teachings of CAAD but due to wider social and technological changes. Computerization of professional activities and entertainment has been playing a more significant role in practice than the academic CAAD training of new professionals. Consequently, CAAD has been redefining its position in the academic and architectural worlds with respect not to internal priorities but to external opportunities presented within the overall framework of computerization. Interest in the issues raised in the 1995 paper decreased as survival became the main priority. Such opportunism is harmful to CAAD. If the field is expected to survive the current period of uncertainty and flourish again in the near future, there is every reason for focusing on its structure, principles, theory and methods. Internal coherence is of paramount importance for performance, clarity and realization of added value. But equally if CAAD is nearing an end, reflection on what has been achieved, what remains to be done and why and how we should attempt it can lay sound foundations for the epigoni of CAAD. Transmitting the legacy of CAAD is not only a scientific obligation but also a structural opportunity for the continuation of fruitful lines. One of the basic problems for doing this is that internal criticism of CAAD has been scarce and fragmentary. Overviews of the field tend to stress the positive and gloss over missing parts and aspects. In the 1995 paper, Maver points out seven fundamental weaknesses that undermine the future or the legacy of CAAD. Regardless of the completeness or accuracy of these sins, we can treat them as a valid departure for an analysis that returns appropriate amends or preferably complementary virtues that can be attained in CAAD.

2. Origins and levels

One of the primary reasons for many of the sins mentioned in the 1995 paper is the mixed origins of CAAD and the resulting divergence and heterogeneity in approaches and goals. Roughly speaking we can distinguish between two main ambitions behind the emergence of CAAD as a scientific field. The first is a lofty one: the automation of design, i.e. the...
replication of cognitive processes, professional knowledge and skills with machine intelligence [3]. Design automation has many flavours, ranging from the symbiosis of the human designer with digital intelligent assistants to the complete substitution of the designer with computerized systems. Many flavours are linked to wider considerations and ambitions, such as user participation in designing. The second ambition is more practical: the development of computer-based means for the representation of architectural designs [4]. Computer-based design representations also come in many variations: two-dimensional, three-dimensional, static, dynamic, interactive, pixel-based, vector-based, compound or multimedia, etc. The most basic form is computer-aided drafting (CAD); the replication of analogue drawings with vector-based computer graphics. CAD has dominated a large proportion of CAAD research and teaching. At a certain time it seemed as if every self-respecting CAAD unit in the world were developing its own CAD system. Even in theoretically-oriented units where design automation was the primary concern, CAD or similar systems were a frequent necessity as a means of implementing and disseminating generative approaches and systems.

However, bringing these two ambitions together is not as straightforward as it might seem. The possible conflicts become apparent when we consider them against the different levels a computational process. In recent years it has become commonplace to distinguish between three interrelated levels [5, 6]:

1. Theoretic or semantic: a general, sometimes abstract specification of the process in a way that allows computational analysis and implementation
2. Algorithmic or syntactic: a more detailed specification of the different strategies that can be followed within the constraints of the computational theory, down to the processing steps to be taken and their sequences
3. Implementational or physical: the actual programming of the process on a particular computer using a particular language

For example, when the three levels are applied to the making of a calculator, mathematics is at the level of the computational theory, the algorithms specify how the different calculations are performed, and the implementation deals with the hardware and the software that is used to apply the algorithms. Despite local vagueness and partial overlaps, such distinctions help reduce complexity and support focused handling of particular problems without ignoring the overall framework of the process.

It must be stressed that the analysis of a process into the above three levels tends to put emphasis on the level of the computational theory and,
to a lesser extent, to the algorithmic level. Implementation is frequently treated as irrelevant or neutral. This tendency is frequent in areas such as artificial intelligence, cognitive psychology, and cognitive science. Nevertheless, even in such areas there have been alternative views that propose the implementational level as primary over the theoretic and algorithmic, e.g. that cognitive models should be assembled by making extremely detailed replicas of neurons [7].

Regardless of origin CAAD has traditionally emphasized higher-level explanations of design processes, as exemplified by the tenet that the ‘D’ in CA(A)D stands for ‘design’ rather than ‘drawing’ or ‘drafting’. Despite the substantial demands of the implementational level and the resulting effort and expense, CAAD has sought justification of its principles and products in theory and method. Even straightforward applications of computer graphics had to have a theoretic or methodical foundation. In many cases this has returned merely tentative associations but connections between theory, method and implementation have also provided selection criteria and application constraints for general-purpose techniques. Moreover, in-depth analyses of the particularities of the architectural domain have resulted into innovative applications [8], while techniques originally conceived as methodical solutions of practical problems have developed further into widely applicable formalisms by virtue of the emphasis they placed on the algorithmic or methodical level [9, 10].

The unifying influence of the hierarchy of these three levels should not obscure fundamental problems that arise from differences in the treatment of the same subject matter. In the field of CAAD differences in origin mean that the same activities, processes and representations are frequently approached from different viewpoints. If the particular viewpoint dominates over the overall framework of architectural computerization, it is not surprising to see very partial and elliptical treatments of fundamental design ideas or instruments. These may not only ignore other approaches and corresponding priorities but also undermine aspects and activities that should by any means be considered as complementary.

Differences in origin may also result into strange, unproductive demarcations. Developments in design automation can be indifferent to practical applications. This returns results that seldom transgress the limitations of an abstract demonstration and develop into prototypes with a wider meaning and applicability. Lack of connections with conventional practice may mean that too many aspects are neglected to make computational design theories useful even for the algorithmic / methodical level. Consequently, design automation remains an academic subject with introverted treatises and prescriptive methods as main products. Reversely, emphasis on (usually commercial) CAD or visualization systems often shows disregard for a comprehensive analysis of professional
requirements that could return usable theories and methods. The result is a superficial replication of conventions and routines that perpetuates current problems and limitations. Theoretical or methodical justifications are usually heavily biased, prescriptive and speculative.

Closer examination of CAAD overviews reveals attempts at a forced coherence on the basis of tentative associations between theories, with reference to schematic views of the design process that provide pigeonholes for each issue, theory and method, or commonsensical taxonomies of imported techniques. Even worse, many overviews have an ephemeral character based on bandwagons: CAAD has been guilty of shifts from so-called paradigm to so-called paradigm and even of elevating implementation mechanisms, metaphors and similes to the level of computational theory. Such shifts may have reinforced links to other disciplines (primarily computer science) and facilitated technology transfer but have also reduced attention for intrinsic issues and relevance to practice.

3. Sins and virtues

In this framework the 1995 paper is a useful summary that reduces the complexity of internal differences and external relations to a conveniently small set of deadly sins. This characterization is an effective reference to a tradition that derives from the early days of the Christian church. Church fathers apparently attempted to encapsulate their teachings into compact mnemonic devices that would help lay, frequently illiterate people understand the Christian faith and apply it to their daily life. The seven deadly sins were codified by Pope Gregory I at the end of the sixth century but were amended as late as the seventeenth century [11, 12]:

1. Pride or vanity
2. Envy
3. Gluttony
4. Lust
5. Anger or wrath
6. Greed or avarice
7. Sloth

The sins represent common transgressions of both the spirit and the flesh. Only the first one (pride) refers to a direct relationship with God. The second, fourth and fifth (envy, lust and anger) have to do with the relationship with other humans. The order of the sins has to do with their seriousness. Originally Pope Gregory I had arranged them as follows (from most serious to least serious offences against divine love): pride, envy, anger, sadness (later replaced by sloth), avarice, gluttony, and lust.
The 1995 paper proposes seven different deadly sins for CAAD:

1. Macro-myopia, a term coined by Paul Saffo of the Institute of the Future in Palo Alto in 1994 to describe overzealous claims resulting into overestimations of the short term impact and underestimations of the longer term impacts
2. Déjà vu, i.e. repetition of earlier work without building on it
3. Xenophilia: the obsession with importing concepts and procedures from other disciplines
4. Unsustainability, resulting from effort being devoted to facilitating the practice of architecture with correspondingly less attention given to achieving design solutions which yield improved quality to the building client and user
5. Failure to validate research results
6. Failure to evaluate the usability and functionality of research products in teaching or practice
7. Failure to criticize ourselves and our peers

A tentative correlation between the two lists of sins is possible if we choose to ignore the differences between the material and the spiritual (Table 1). This correlation obviously does no justice to Maver’s criticism and is variably based on relations between cause and effect (e.g. pride and déjà vu), direct correspondence (e.g. envy and xenophilia) and analogy (e.g. gluttony and unsustainability):

<table>
<thead>
<tr>
<th>Christian sins</th>
<th>CAAD sins</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pride</td>
<td>Déjà vu</td>
</tr>
<tr>
<td>Envy</td>
<td>Xenophilia</td>
</tr>
<tr>
<td>Gluttony</td>
<td>Unsustainability</td>
</tr>
<tr>
<td>Lust</td>
<td>Failure to criticize</td>
</tr>
<tr>
<td>Anger</td>
<td>Failure to validate</td>
</tr>
<tr>
<td>Greed</td>
<td>Macro-myopia</td>
</tr>
<tr>
<td>Sloth</td>
<td>Failure to evaluate</td>
</tr>
</tbody>
</table>

Table 1: Sins compared

Probably the main function of this comparison is to facilitate correlation with the seven heavenly virtues that form the counterpart to the Christian deadly sins:

1. Faith
2. Hope
3. Charity
4. Fortitude
5. Justice
6. Temperance
7. Prudence
These can be tentatively associated to the deadly sins, as amends or remedy (Table 2):

<table>
<thead>
<tr>
<th>Christian sins</th>
<th>CAAD sins</th>
<th>Heavenly virtues</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pride</td>
<td>Dijà vu</td>
<td>Faith</td>
</tr>
<tr>
<td>Envy</td>
<td>Xenophilia</td>
<td>Charity</td>
</tr>
<tr>
<td>Gluttony</td>
<td>Unsustainability</td>
<td>Temperance</td>
</tr>
<tr>
<td>Lust</td>
<td>Failure to criticize</td>
<td>Justice</td>
</tr>
<tr>
<td>Anger</td>
<td>Failure to validate</td>
<td>Hope</td>
</tr>
<tr>
<td>Greed</td>
<td>Macro-myopia</td>
<td>Prudence</td>
</tr>
<tr>
<td>Sloth</td>
<td>Failure to evaluate</td>
<td>Fortitude</td>
</tr>
</tbody>
</table>

The implication of Table 2 is that the heavenly virtues can also be considered as a constructive, positive reaction to the CAAD sins. Before doing so, we should also consider the positioning of the CAAD sins on the levels of a computational process (Table 3):

<table>
<thead>
<tr>
<th>Theoretic</th>
<th>Algorithmic</th>
<th>Implementational</th>
</tr>
</thead>
<tbody>
<tr>
<td>Macro-myopia</td>
<td>+++</td>
<td></td>
</tr>
<tr>
<td>Dijà vu</td>
<td>++</td>
<td></td>
</tr>
<tr>
<td>Xenophilia</td>
<td>+++</td>
<td></td>
</tr>
<tr>
<td>Unsustainability</td>
<td>++</td>
<td>+</td>
</tr>
<tr>
<td>Failure to validate</td>
<td>++</td>
<td>+</td>
</tr>
<tr>
<td>Failure to evaluate</td>
<td>+</td>
<td>++</td>
</tr>
<tr>
<td>Failure to criticize</td>
<td>+</td>
<td>+</td>
</tr>
</tbody>
</table>

What emerges from Table 3 is a clear distinction between CAAD sins that concern the higher levels and lower levels. For example, macro-myopia and xenophilia concern almost exclusively the level of computational theory, while the failure to validate and evaluate focus variably on the algorithmic and implementational level. The notable and obvious exception is the failure to criticize, which applies to all levels. Table 3 can be used in two ways. The first is as a basis for the ordering of sins by their importance. This does not necessarily mean that one should follow the hierarchy of the three levels of a computational process. Assigning priority to implementation or algorithms is equally acceptable. The second application of Table 3 is the clustering of sins and virtues by the three levels. This reveals the dangers and possibilities inherent in each level.

4. CAAD virtues

The applicability of the seven heavenly virtues to CAAD is by definition questionable. Mapping the troubles and necessities of a still evolving field onto an external, arbitrary structure cannot be expected to return a usable classification or a comprehensive collection of direct solutions. All that can be expected is a first impression of the character general notions of virtue take in the field of CAAD and in particular whether a virtue has already
been attained in CAAD. Consequently, in contrast to the 1995 paper, I have not attempted to adapt the seven heavenly virtues to CAAD.

4.1. Faith

Faith is the opposite of pride and a possible solution to the CAAD sin of déjá vu. Conviction and fidelity to fundamental principles and well-defined theories and methods promote continuity and consistency. Faith can also offer a partial solution to xenophilia by moderating the transfer of external knowledge and technology. By propagating the constraints of CAAD theory and practice to related disciplines and fields it is possible to select appropriate transfer candidates and link them directly to their application area.

The main prerequisite to faith is the development of coherent, comprehensive, consistent and relevant theories. Paradigm-hopping and tensions between the different origins of CAAD have done much to impede this. While one cannot be too optimistic about the near future, it is certainly advisable to consider carefully the subject of CAAD faith and choose the constituents and backgrounds of CAAD theory with conviction, transparency and founded argumentation.

Still, examples of faith can be found in CAAD. Most relate to theoretically-motivated systems that have been reached an advanced level of development already in the early period of the field. Faith to the underlying theories has led to extensive and intensive exploration that provides a comprehensive picture of the theories’ possibilities and limitations. A different form of faith refers to continuity in research lines that may lack an explicit theory but nevertheless exhibit a high degree of consistency. The few individuals and even fewer groups have managed to do so have provided us with some of the finest results of CAAD.

4.2. Hope

I have linked hope to anger and to the failure to validate in order to make the following point: continuity in research is based on well-founded and well-defined expectations. The formulation of assumptions and validation approaches is not an externally imposed obligation but an integral part of a sincerely motivated research. Maves’ criticism on the lack of validation in CAAD is certainly true but in many cases it is a matter of methodical poverty. CAAD is still in need of reliable validation devices that replace the simplistic notion of developing systems that can be used experimentally, in education or in practice.

4.3. Charity

Xenophilia can be construed as a form of envy. Importing alien concepts and procedures is frequently motivated by the desire to achieve in CAAD what
such concepts and procedures have achieved elsewhere. Charity contradicts envy by stressing the importance of giving. CAAD has certainly been guilty of xenophobia but has equally been generous with the dissemination of its results and products in architecture. The motives for doing so have not always been altruistic but I do not think that CAAD can be accused of not having been helpful with architectural automation both in education and in practice.

4.4. Fortitude

Fortitude is a clear opposite to sloth and a remedy for the failure to evaluate. A common problem in academic research is that it tends to stop when its results become truly interesting and applicable in practice. The few results that manage to cross over to practice seldom receive the attention they deserve, especially in terms of analysis, evaluation and feedback. Bridging the gap between the two requires strength, courage and endurance. Unfortunately such characteristics are frequently lacking in CAAD research. Many demonstrations developed rather effortlessly in CAAD fail to develop into usable prototypes because they miss the guidance rigorous and consistent evaluation can provide.

4.5. Justice

Justice may be only loosely related to lust but is an appropriate antidote to the failure to criticize. CAAD research frequently degenerates into an innovation race devoid of self-criticism and of constructive criticism of each other’s performance and achievement. As a result, CAAD has an underdeveloped sense and structure of impartial and fair criticism. Analyses of precedent research tend either to pick only the elements that are useful for the development and defense of a particular project or consider what has been achieved previously through the prism of a very specific approach. This is further accentuated by the general lack of validation and evaluation.

Having said that, one of my greatest joys in reading CAAD papers is the comprehensive and dispassionate manner in which a few researchers consider precedent research. In just a few pages they manage to provide an overview of work done in the field, relations to other disciplines and criteria for their evaluation and classification. If this example is also followed in overviews of the field (instead of the customary propagation of expectations) we can expect a sound foundation for the application of existing results, as well as for further research.

4.6. Temperance

Temperance can be viewed as the opposite to gluttony as well as of greed and lust. Its link to unsustainability refers to the overemphasis on design approaches and generative systems in CAAD. This is probably symptomatic...
of a wider tendency in architecture (especially since postmodernism). Issues such as the production of complex forms and the exhaustive generation of classes of designs seem to outweigh the necessity for reliable projections of building behaviour and performance or design guidance on the basis of such projections. Technologies such as visualization and simulation are more often used in a decorative sense than for the analysis, evaluation and communication of designs.

Research into computational design analysis probably contains the best examples of temperance in CAAD. The analytical power of the computer can complement human creativity in an unobtrusive, constructive manner that not only helps the improvement of a design but also focuses attention on primary, universal issues in the built environment (as opposed to the self-centered concerns of a group or even a whole profession).

4.7. Prudence

Prudence can be a solution to macro-myopia and its underlying greed, which too is arguably symptomatic of a wider framework – this time, computing. Computing and by extension CAAD can be exuberant and inspiring but also naive in their perception of application areas and downright cynical in their quest of an early market share. This last characteristic is a particularly worrying cause of macro-myopia. It leads to wrong estimations not because of exaggeration or unreliable data but purely out of greed.

Confronting greed and macro-myopia is probably the toughest challenge for CAAD. In a landscape of radically evolving technologies and related social conditions we are asked to remain vigilant and thoughtful, to consider carefully the requirements and potential of architectural computerization so as to establish an appropriate pace of development. But even when this is achieved, it is quite hard to neutralize all external influences and the resulting false promises. Once again CAAD is betrayed by its origins and sources.

5. Conclusion

When I started considering the relationship between sins and virtues in CAAD I had expected to transform the criticism of the 1995 paper into positive guidelines for the future with a firm basis on current practices and achievements. I had not anticipated that my conclusions would share the same mildly pessimistic undertones of the 1995 paper. This is not due to the lack of worthy examples of the virtues to be attained. CAAD has already managed to produce a substantial corpus of intelligent, relevant theories, methods and techniques. Most are not directly applicable in practice or even in education, but nevertheless form a firm foundation for the further development of the field.

The main problem appears to lie in the climate surrounding CAAD. The ongoing democratization of the computer forces the pace of developments
in the field and may have far-reaching consequences. Given the different origins and diverging priorities and approaches of CAAD it is not inconceivable that the field will deconstruct and that a part of it will evolve into a specific approach in architectural theory, another will provide computing assistance to other architectural and building specializations, and yet another will focus on the computational dimension of different application areas that will absorb it.

Any attempt to define CAAD in this climate involves too many aspects and issues for a straightforward, succinct answer to critical questions such as the ones raised in the 1995 paper. Moreover, the way such questions are stated and structured is not without significance. Reference structures such as the three levels of computational theory, algorithm and implementation assist in the analysis and interpretation of the strengths and weaknesses of the field – its sins and virtues – towards the wide, in-depth discussion proposed in the 1995 paper.

Apologies

Originally I intended to provide actual examples of each of the proposed virtues from CAAD literature. I actually started doing so but soon I realized that it was pointless. The only way I could do justice to the field was to include every relevant publication in the references. Obviously this is beyond the scope of this article and ultimately trivial. I must therefore apologize to everyone who deserves having his or her name and work linked to the virtues of CAAD for not being able to publish these links.

References


