Automated Simulation and Study of Spatial-Structural Design Processes

Juan Manuel Davila Delgado[1], Herm Hofmeyer[2]
Eindhoven University of Technology, Netherlands
[1]http://www.tue.nl/en/university/departments/built-environment/the-department-of-the-built-environment/staff/detail/ep/e/d/ep-uid/20104850/
[2]http://www.tue.nl/en/employee/ep/e/d/ep-uid/19941803/ep-tab/3/
[1]j.m.davila.delgado@tue.nl, [2]h.hofmeyer@tue.nl

Abstract. *A so-called "Design Process Investigation toolbox" (DPI toolbox), has been developed. It is a set of computational tools that simulate spatial-structural design processes. Its objectives are to study spatial-structural design processes and to support the involved actors. Two case-studies are presented which demonstrate how to: (1) study the influence of transformation methods on design instances and (2) study the influence of transformation methods on the behavior of other transformation methods. It was found that in design instances with the same type of structural elements the influence of a specifically varied transformation method is more explicit; while, when different types are present this influence is more undetermined. It was also found that the use of two specifically different structural modification methods have little influence on the sub-sequential spatial transformation method.*

Keywords. *Design process research; design process simulation; spatial design; structural design.*

INTRODUCTION

In the Architecture, Engineering and Construction (AEC) field, design processes are complex and multidisciplinary undertakings in which designers and engineers work together on the same problem to come up with feasible solutions. The final solution is usually the result of a cyclic process, in which the initial solution undergoes several changes and adaptations to meet pre-defined and arising requirements.

It is assumed that by improving the design process, the design outcomes will improve as well (Cross, 2008; Brooks, 2010; Kalay, 2004). Consequently, efforts have been carried out on the research of design processes, roughly subdivided in two categories: (1) the development and study of design models, which is the formulation of frameworks to organize the process of designing and (2) the generation of support methods or tools to aid in the design process. In the last category computational tools have been developed to increase productivity (Grobman et al., 2010), to ease the communication and the exchange of information between parties within the design process (Haymaker et al., 2004) and to take an active role on the design process and generate design solutions (Shea et al., 2005). However, little research has been carried out in which the computer is used to study the design process itself (Kalay, 2004; Coates, 2010).

The objective of the project presented in this paper is to increase the knowledge on spatial-structural design processes and consequently to support the involved actors. To that end a computational toolbox, a so-called “Design Process Investigation” (DPI toolbox) has been developed. More concretely, the DPI toolbox as presented in this paper, seeks to fulfill the following two aims: (1) to study the influence of a selected transformation method on design instance evolution; and (2) to study the influence of a selected transformation method on the behavior of the other transformation methods. The next section will briefly explain the DPI toolbox. Then a demonstration of the types of investigations which can be performed is shown, and lastly a short discussion and an outlook on further work are presented.

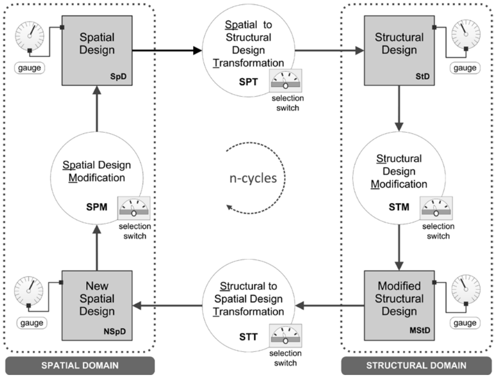


Figure 1
Design Process Investigation toolbox framework.

DESIGN PROCESS INVESTIGATION TOOLBOX

The DPI toolbox framework (Figure 1) prescribes specific and identifiable steps to reach a design solution. In that sense it could be categorized as a prescriptive design model (Cross, 2006). However, the objective of prescriptive design models is to ensure successful and consistent results; whereas the objective of the DPI toolbox is to simulate design processes so its outcomes and more importantly the process itself can be studied.

Design processes are cyclic and multidisciplinary tasks where both design solutions and design requirements undergo changes and adaptations before a definitive solution is achieved (Maher et al., 1996; Haymaker et al., 2004). Also, design requirements are usually “ill-defined” and the design process is often not recorded properly, so it is difficult to trace back or investigate the process later on. The DPI toolbox framework is developed to address those characteristics and problems of a design process.

The DPI toolbox framework defines the process to be followed. During this process a design instance is subject to four different transformation phases acting within or between the spatial and structural domains. It works as follows (Figure 1): First, a Spatial Design (SpD), in the spatial domain, is transformed into a Structural Design (StD) in the structural domain. Then, within the same domain, the StD is altered into a Modified Structural Design (MStD). After that, the MStD is transformed into a New Spatial Design (NSpD) that finally is altered into a Modified New Spatial Design (MNSpD), completing one full cycle. This cycle can be repeated causing the spatial and structural design instances to co-evolve. For co-evolutionary designs, no classical convergence criteria can be used to stop the process; but, if requirements (spatial design instances) and solutions (structural design instances) do not change anymore a (local) optimum is believed to be found (Maher et al., 1996).

Two other relevant characteristics of the DPI toolbox framework are: the “transformation and modification selection switches” and the “gauges” (Figure 1). These components have the objective of facilitating the study of the simulated design processes. The idea is to use the DPI toolbox to simulate different design processes, each with different transformation procedures, and to measure the resulting design instances, by the gauges, through the cycles for later comparison. In this way, it is possible to study the influence of transformation procedures on design instances and on sub-sequential transformation procedures.

Note that the DPI toolbox framework only prescribes the existence of a set of transformations, relations, and measurements (by the gauges) between two different domains within a cyclic design process; it does not define specific transformation or

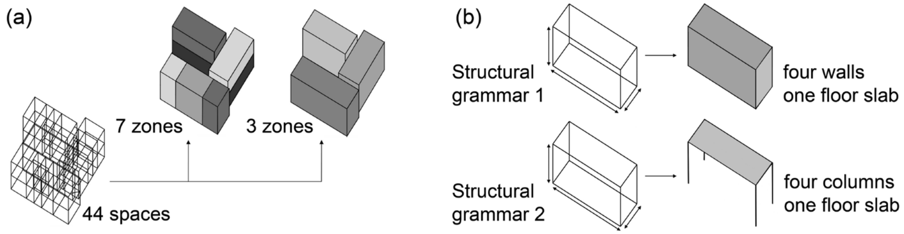


Figure 2
(a) Example of the zoning algorithm; (b) Two structural grammars used in the DPI toolbox.

measurement procedures. Thus, the selected transformation and measurement procedures used in the DPI toolbox are not unique in any way, and these used in this paper were chosen primarily for their availability. The procedures could be changed in the future to further study spatial-structural design processes.

As mentioned before, the DPI toolbox consists of four transformation phases and these phases will now be shown to consist of several stand-alone procedures, put together in a seamless process. Some of the used procedures have been widely studied and utilized in the AEC field, e.g. shape grammars, pattern recognition, and FEM simulations; others have been developed specifically for the DPI toolbox, e.g. geometrical redefinition and kinematic stabilization. Next, the four phases of the DPI toolbox, as implemented, will be briefly described.

Spatial to Structural Design Transformation (SPT)

The first phase generates a structural design instance and performs a FEM simulation with it, all based on the spatial design instance as used for input. The generated structural design only intends to formulate a starting point for the design cycles, and it does not intend to be an immediate optimal solution for the inputted spatial design. Likewise, the FEM simulation is not meant for stress engineering, but is merely used to give an indication of the structural behavior of the proposed structural design.

The Spatial Design consists of a set of volumes or "spaces". So far, the DPI toolbox is restricted to work with right cuboids, parallelepipeds bounded by six rectangular faces, so that each adjacent face meets at a right angle. Furthermore, the right cuboids or spaces should be aligned with the global coordinate system. The Spatial Design undergoes several transformations by procedures that are grouped in the following categories: (a) proposal of the structural design, (b) preprocessing, and (c) structural calculations.

The proposal of the structural design consists of two procedures: first structural zones are created and then, based on them, structural elements are generated. For the first procedure, the DPI toolbox uses an in-house developed automated 3D zoning algorithm (Hofmeyer and Bakker, 2008) (Figure 2a). It defines structural zones (elementary structural entities) based on sets of spaces. This procedure subdivides the Spatial Design into a number of zones, (grouped spaces) and these are used as a basis to generate structural elements. For the next procedure, structural grammars (Shea and Cagan, 1999) are used to generate structural elements. Structural grammars resemble shape grammars used in the AEC area (Stiny, 1980). They prescribe which structural elements can be used depending on the geometrical properties of the previously generated zones (Figure 2b).

Regarding (b) the preprocessing category, once a structural design has been generated, it has to undergo several procedures to be able to be simulated by a Finite Element Method (FEM). First, the geometry of the structure has to be redefined to ensure that all the finite element nodes will be coincident and to determine the wind loaded surfaces. Then the structure should be made kinematically determined, loads and constraints should be applied, and a meshing algorithm has to be performed.

Lastly, regarding (c) the structural calculations, the following procedures should be mentioned: A first-order linear elastic FEM simulation is carried out to predict nodal displacements in the structural design; then, the strain energy of each finite element is calculated. A clustering algorithm groups the finite elements into clusters based on their strain energy and a color-coded visualization is generated. The data obtained during this step will be the basis for the next phase's procedures, presented below. More information on this procedure can be found in (Hofmeyer and Davila Delgado, 2013).

Structural Design Modification (STM)

Having generated a Structural Design, the next step is to improve its structural behavior. Even though the procedures implemented in this phase follow closely those of traditional structural optimization, their objectives are slightly different. The objective of this phase of the DPI toolbox is not to obtain an optimal structural design per se, as in the traditional way, but to modify the structure only into the direction of an optimal design. Thus this phase is called Structural Design Modification rather than optimization.

This structural modification is based on minimizing strain energy. A structure that deforms under a given case of loads and constraints shows strains in its finite elements. The amount of strain energy in a finite element is a measure of its participation in bearing the applied loads. So, finite elements showing low strain energy can be regarded as being under-utilized and thus may be deleted. Two versions of existing structural optimization methods have been implemented in the DPI toolbox namely: Evolutionary Structural Optimization (ESO) and Topology Optimization (TO). A detailed explanation of this phase can be found in (Hofmeyer and Davila Delgado, 2013). Note that the version of ESO used has been modified so that only a single iteration is run in the optimization procedure (in this paper referred to as 1ESO). This is done because accurate enough results can be obtained and computation time is reduced.

Structural to Spatial Design Transformation (STT)

In this phase the MStD, an arrangement of structural elements, is transformed into the NSD, an arrangement of spaces. This is currently implemented as follows: First, it is indentified which finite elements have been deleted in the previous phase and to which space from the inputted Spatial Design they belong to (i.e. which deleted elements are contained within which space). Based on that information the spaces that contain many deleted (under-utilized) finite elements are removed. In other words, spaces that contain less elements contributing to withstand the applied loads, are in a structurally-seen less important zone, and are thus deleted (Hofmeyer and Davila Delgado, 2013).

For the current implementation, the first 30% of spaces with most deleted elements are removed, and then the remaining spaces are investigated. If spaces with the same number of deleted elements as the already removed spaces exist, they are removed as well. Note that in almost virtual case that all spaces have the same number of deleted elements, then only the first listed 30% of the spaces are deleted. This implementation is referred in this paper as "*Delete Spaces*".

Spatial Design Modification (SPM)

In this process, the NSpD will now be modified into a MNSpD that will serve as the input for a next cycle of the DPI toolbox. The main objective of this phase is to modify the NSpD for the next cycle such that at least some of the properties of the SpD, which may have disappeared during the transformations of the cycle, are restored. For example, in the end of the previous phase, spaces were deleted from the SpD and thus the NSpD has less volume and fewer spaces. Therefore, in this phase, the NSD could be scaled up to the same volume as the SpD and then some spaces within the NSD could be subdivided in order to restore the initial number of spaces. This phase is explained in more detail in (Davila Delgado and Hofmeyer, 2013) and it is referred to in this paper as "*Re-scale and Subdivide*".

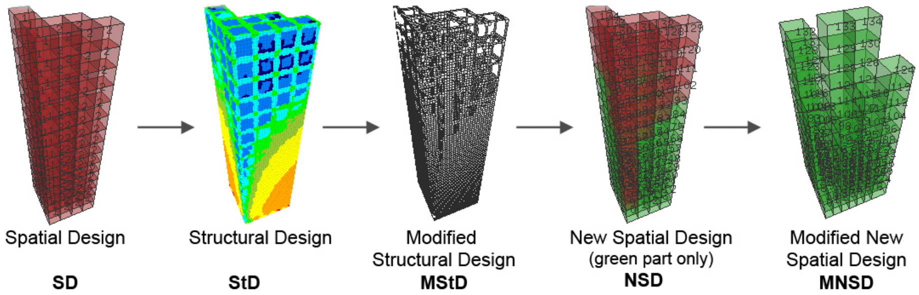


Figure3
A typical DPI toolbox run.

DPI toolbox example run

Figure 3 shows a typical run of the DPI toolbox. Starting from left to right: the Initial Spatial Design; the Structural Design, here displaying its strain energy distribution; the Modified Structural Design where the under-utilized elements have been deleted; the New Spatial Design (green part only) where the spaces with more under-utilized elements (red) have been removed; and the Modified New Spatial Design, which has the same volume and number of spaces as the initial spatial design.

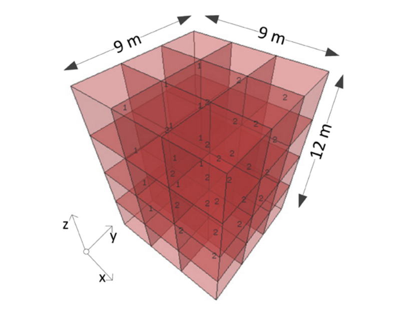
DEMONSTRATION

The main purpose of this section is to exemplify the types of investigations that can be performed with the DPI toolbox. Note that the cases presented here serve as a proof of concept and that in further stages of the research real-life and more complex case-studies will be performed.

Two case-studies are presented to demonstrate aims (1) and (2) as presented at the end of the Introduction section. In Case-study I, it is investigated how a change of the transformation method (a different structural grammar in this case) influences the evolution of the design instances (in this case structural designs) through the cycles. In Case-study II, it is investigated how a change of a transformation method (in this case 1ESO vs. TO for STM) influences the behavior of the sub-sequential transformation method (STT), again with respect to an observation through the cycles.

Figure 4 shows the initial Spatial Design used for both case-studies and the defined settings of the DPI toolbox respectively. For each case-study, two simulations (runs) have been performed, consisting of four cycles each.

Figure 4
Initial Spatial Design used for case-studies I and II; Tables list the respective DPI toolbox settings.



Spatial Design: 4 levels consisting of a 3x3 spaces as ground plan

*Number of clusters of finite elements to be deleted in STM.
**Divisions in which the structural elements will be subdivided.
***One zone is created for each space.

Case-study I	
I-A:	4 cycles
SPT:	
Zoning***	1space is 1zone
Grammars	**1roof-slab and 4 walls**
Loading	Gravity load
Divisions**	8
Deleted Clusters*	2
STM:	1ESO
STT:	Delete Spaces
SPM:	Re-scale and Subdivide
I-B:	4 cycles
SPT:	
Zoning***	1space is 1zone
Grammars	**1roof-slab and 4 columns**
Loading	Gravity load
Divisions**	8
Deleted Clusters*	2
STM:	1ESO
STT:	Delete Spaces
SPM:	Re-scale and Subdivide

Case-study II	
II-A:	4 cycles
SPT:	
Zoning***	1space is 1zone
Grammars	1roof-slab and 4 walls
Loading	Gravity load
Divisions**	8
Deleted Clusters*	2
STM:	**1ESO**
STT:	Delete Spaces
SPM:	Re-scale and Subdivide
II-B:	4 cycles
SPT:	
Zoning***	1space is 1zone
Grammars	1roof-slab and 4 walls
Loading	Gravity load
Divisions**	8
Deleted Clusters*	2
STM:	**TopOpt**
STT:	Delete Spaces
SPM:	Re-scale and Subdivide

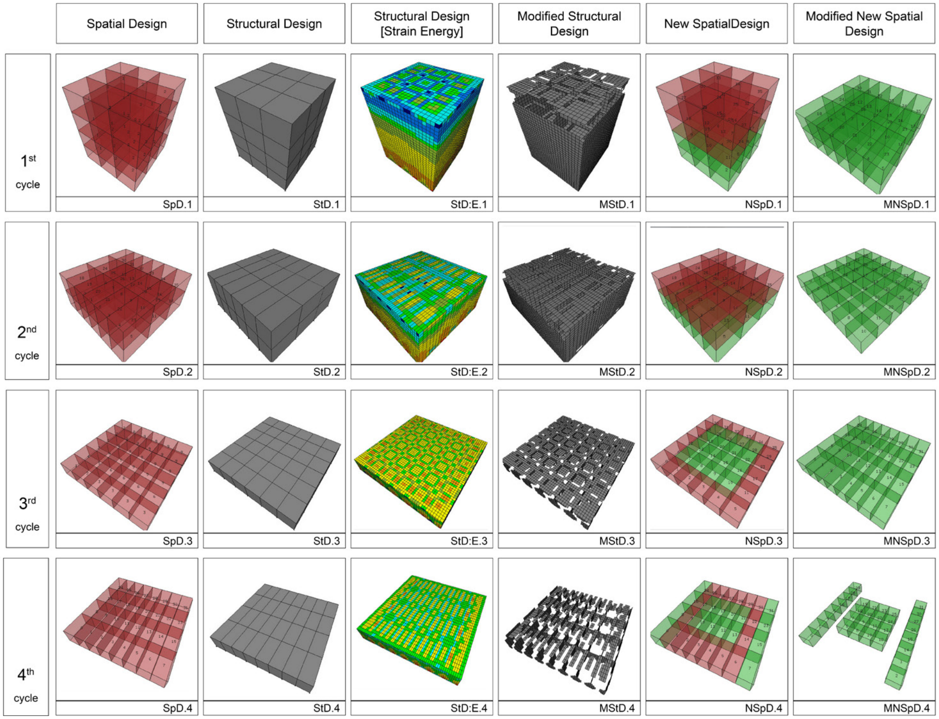


Figure 5
Resulting design instances for run I-A (which are the same as for II-A).

Case-study I

For Case-study I, two different runs have been performed: I-A and I-B, with structural grammars 1 and 2 (Figure 2b) respectively. All other settings were kept the same (Figure 4, table: Case-study I). Figure 5 and 6 show the resulting design instances of both runs (Note that the resulting design instances of case-study I-A are the same as those of II-A, so both are presented in Figure 5). In each figure, each row presents the results of one cycle, while the columns represent a phase within each particular cycle. Figure 7 presents the two measurements taken in each cycle for the design instance under investigation: the maximum nodal displacement (d_{max}) and the total strain energy (U_t). These measurements have been selected because they give an indication of the structural behavior of the resulting StD. A graph of each measurement through the four cycles is also presented in Figure 7.

For run I-A, it can be seen that d_{max} and U_t decrease at approximately the same rate at every cycle (Figure 7). This is probably due to the decrease of the number of building levels through the cycles. In Figure 7, on the top right corner of graph **b**, the decrease of levels through the complete run is plotted. That rate is similar to the rate of d_{max} and U_t. So it is not unlikely that there is a link between the number of levels in a StD and its d_{max} and U_t. Note that the spatial design instances in each cycle have approximately the same volume, number of spaces, and structural elements and that only dead load has been used as a load case. So, even though the structural mass of all the design instances is quite similar (Figure 7, table: Case-study I-A) -meaning that the total amount of load is quite similar as well- U_t is different. An explanation for the behavior above is that

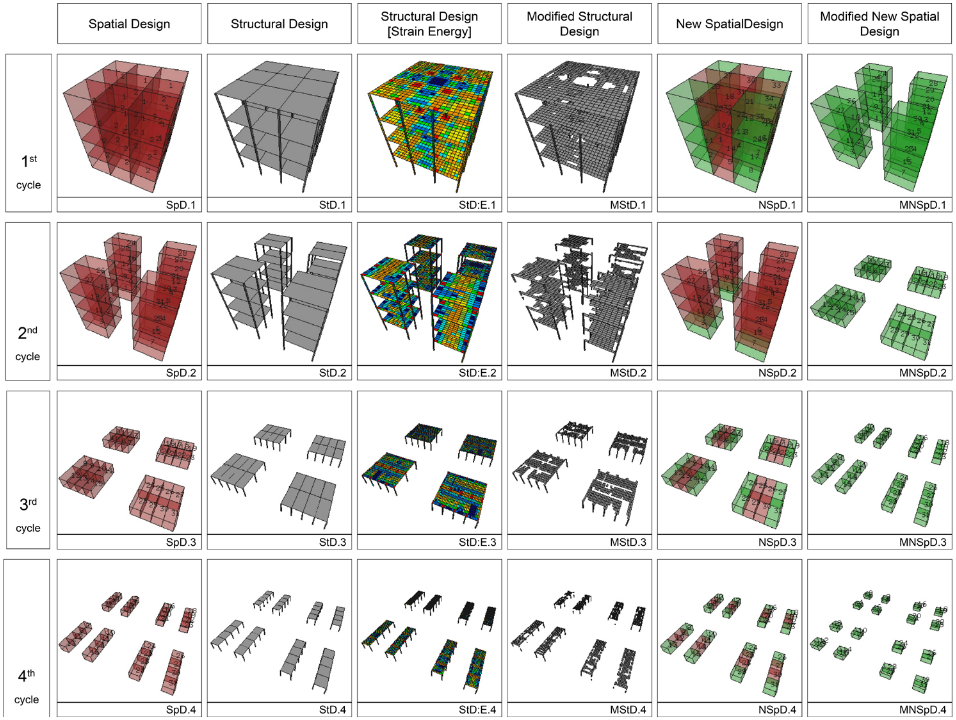


Figure 6
Resulting design instances for run I-B.

in design instances with several levels the structural elements at the lower part of the structure have to withstand their own weight plus the weight of the structural elements on top of them and thus show higher strains.

In the last cycle the design instances have the same number of levels as in the previous cycle. Consequently U_t does not reduce significantly, but d_{max} does. This is because the horizontal structural plate elements that form StD.4 are rectangular, instead of square, and such elements tend to deform more.

For run I-B, using a different structural grammar, the evolutions of d_{max} and U_t follow the same pattern; but they do not correspond so clearly to the evolution of the number of levels, as in run I-A. In Figure 7 it can be seen that d_{max} and U_t increase seriously after the first cycle, even though the number of levels remains the same, and then decrease in each subsequent cycle. The initial increase can be explained by two reasons: (1) after the first cycle the design is divided into four fragments. In these fragments fewer columns have to support more roof-slab area and (2) the roof-slabs in StD.2 are rectangular, which deform more than square types. In both runs, I-A and I-B, d_{max} is always located at middle of the highest roof-slab so their dimensions (ratios) have a high influence on d_{max} and U_t. The second cycle's decrease could be explained due to the decrease in the number of levels, as observed in the previous run. Finally, the last decrease is due to the square shape of the resulting roof-slabs which deform less and thus yield less U_t.

In summary, during the evolution of run I-A a continuous decrease for d_{max} and (partly) for U_t can be observed. This decrease is directly linked to the number of levels of the design. Conversely, in the

		I-A: Flat-shells				I-B: Columns			
Cycles		1	2	3	4	1	2	3	4
Maximum displacement	d_{max}	0.179604	0.0719628	0.0623938	0.0571595	0.348749	1.31948	0.732043	0.490521
Direction		uz	uz	uz	uz	uz	uz	uz	uz
Structural mass	m	178.2	169.4	162.8	142.7	65.8	74.5	64.8	70.15
Number of Levels	L	4	2	1	1	4	4	1	1
Total strain energy	U_t	244166	93140.5	41947.5	40565.6	80937.5	338001	203717	134220

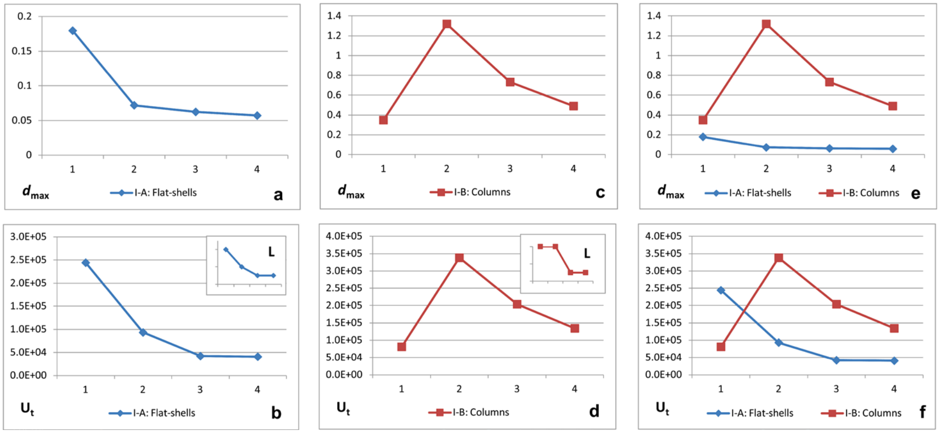


Figure 7
Result tables and graphs of runs I-A and I-B.

evolution of run I-B, no pattern can be recognized. This might be explained because in run I-A all the structural elements are the same; whereas for run I-B it is a mixture of flat-shells and columns.

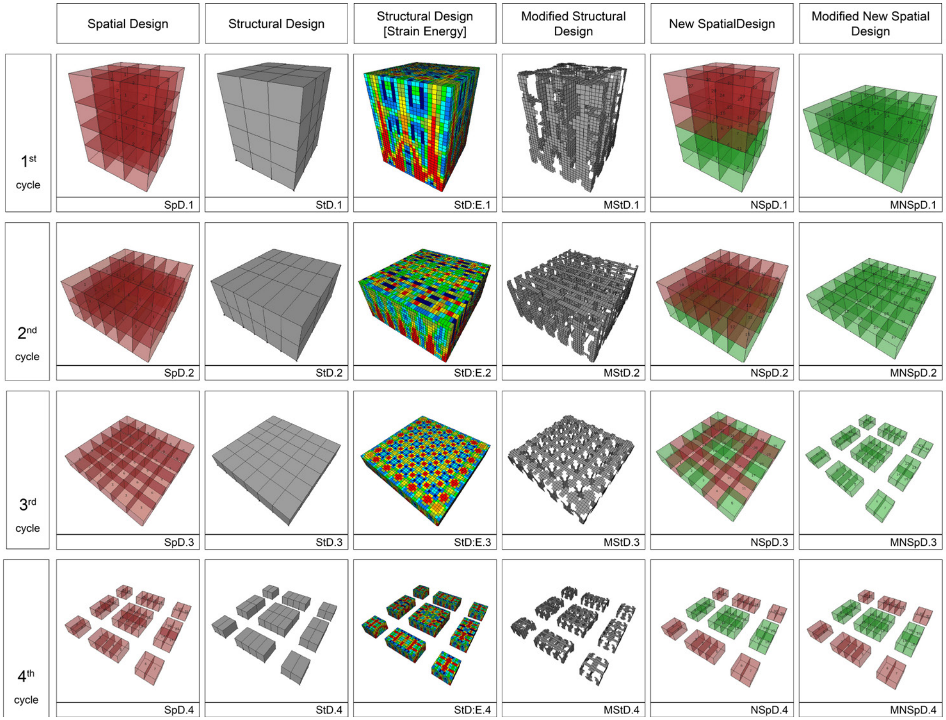
Case-study II

Also for case-study II, two different runs have been performed: II-A and II-B, using 1ESO and TO for STM, respectively. All the other settings were kept the same (see also Figure 4, table: Case-study II). Figure 8 shows the resulting design instances of run II-B. Figure 5 presents the resulting design instances of run II-A, as they are the same as for run I-A. Figure 9 presents the two measurements taken in each cycle: the reduction of U_t and the difference between the number of spaces of the SpD and the NSpD. They were chosen because they are indicators of the performances of STM and STT respectively. Note that the TO procedure optimizes the structural design by decreasing the density of the less strained finite elements and increasing the density of the most strained ones. During this process, a "pseudo-U_t" is used (in fact a strain energy to the power of a certain penalty) which cannot be compared directly with the physically realistic U_t from 1ESO. For that reason the U_t values from run II-A were adjusted. This was done by (a) matching the density of the structural elements in the 1ESO calculations to the density of the first iteration of the TO procedure, and (b) by calculating the energy of the 1ESO calculations taking into account the power of the penalty. In this way, even though the U_t values are not "physically accurate" comparisons between the two procedures can be made.

The results tables of Figure 9 present the strain energy of the structural design before and after the STM procedure is performed, U_t and $U_{t\text{-final}}$ respectively.

Note that the U_t values of both runs are very similar. This is because they both have a similar StD (Figure 5 & 9) with the exception of the last cycle in which the StD -and thus the U_t- differs. Even though for both runs $U_{t\text{-final}}$ decreases at approximate the same rate, $U_{t\text{-final}}$ in run II-B is always lower. This is because TO minimizes U_t, while 1ESO minimizes structural mass, by deleting the structural elements

Figure 8
Resulting design instances from run II-b.



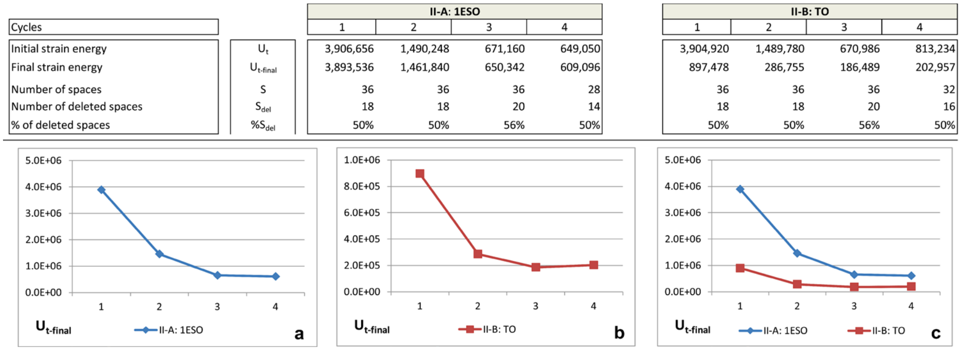
with less U_t. So in 1ESO, U_t is hardly reduced. It is also noticeable that the reduction of $U_{t\text{-final}}$ between two runs diminishes for every cycle. This is because in design instances with more levels U values among finite elements differ more, because due to gravity loads, finite elements at the bottom part of the structure yield more strain than the ones at the top part. So there is more opportunity for optimization in a structure with very dissimilar U values among its elements.

However, it can be seen as well that this difference in performance has little effect on the behavior of the subsequent transformation method (STT). For both runs, the specific spaces and the total number of spaces deleted by STT are the same during the first three cycles and it only slightly changes in the last cycle. Thus it can be said that within the current implementation, a different STM seems to have little influence on the behavior of STT.

DISCUSSION AND FURTHER WORK

The DPI toolbox framework and its current implementation were briefly presented. It simulates spatial-structural design processes to: (1) study the influence of a selected transformation method on design instance evolution; and (2) study the influence of a selected transformation method on the behavior of the other transformation methods. Two case-studies were presented, which illustrate the DPI toolbox's potential to aid in the study of design processes.

The first case-study investigated the influence of using a different structural grammar (a different transformation method) in the evolution of the structural design, via the maximum nodal displacement (d_{max}) and the total strain energy (U_t). It was



Cycles		II-A: 1ESO 1	II-A: 1ESO 2	II-A: 1ESO 3	II-A: 1ESO 4	II-B: TO 1	II-B: TO 2	II-B: TO 3	II-B: TO 4
Initial strain energy	U_t	3,906,656	1,490,248	671,160	649,050	3,904,920	1,489,780	670,986	813,234
Final strain energy	$U_{t\text{-final}}$	3,893,536	1,461,840	650,342	609,096	897,478	286,755	186,489	202,957
Number of spaces	S	36	36	36	28	36	36	36	32
Number of deleted spaces	S_{del}	18	18	20	14	18	18	20	16
% of deleted spaces	$\%S_{del}$	50%	50%	56%	50%	50%	50%	56%	50%

Figure 9
Result tables and graphs of runs II-A and II-B.

found that that in design instances with the same type of structural elements the influence of transformation methods is observed to be more explicit while, when different types are present, the influence is more undetermined.

The second case-study investigated the influence of using different Structural Modification Methods (i.e. 1ESO vs. TO) on the behavior of the subsequent Structural Transformation Method (STT). It was found that even though TO generates better structural designs than 1ESO, this has little effect on the behavior of the sub-sequential STT.

In the future, a further set of rigorous academic and real-life case-studies will be devised to benchmark the DPI toolbox. New transformation methods and amendments to the existing ones will also be implemented to further study the design processes.

REFERENCES

Brooks, FP 2010, *The Design of Design*, Addison-Wesley Professional, Boston, MA.

Coates, P 2010, *Programming.Architecture*, Routledge.

Cross, N 2006, 'Designerly Ways of Knowing', *Design Issues*, 17(3), pp. 49–55.

Cross, N 2008, *Engineering Design Methods 4th ed.*, John Wiley & Sons, Chichester, UK.

Davila Delgado, JM and Hofmeyer, H 2013, 'Research Engine: A tool to simulate and study spatial-structural design processes', *Proceedings of CAAD Futures 2013*, July 3-5, Shanghai, China.

Grobman, YJ, Yezioro, A and Capeluto Guedi, I 2010, 'Non-Linear Architectural Design Process', *International Journal of Architectural Computing*, 8(1), pp. 41–54.

Haymaker, J, et al. 2004, 'Engineering Test Cases to Motivate the Formalization of an AEC Project Model as a Directed Acyclic Graphs of Views and Dependencies', *Journal of Information Technology in Construction*, 9(January), pp. 419–441.

Hofmeyer, H and Bakker, MCM 2008 'Spatial to kinematically determined structural transformations', *Advanced Engineering Informatics*, 22(3), pp. 393–409.

Hofmeyer, H and Davila Delgado, JM 2013, 'Automated design studies: Topology versus One-Step Evolutionary Structural Optimisation', *Advanced Engineering Informatics*, Available online 28 April 2013.

Kalay, YE 2004, *Architecture's New Media: Principles, Theories and Methods of Computer-Aided Design*, MIT Press.

Maher, M L, Poon, J and Boulanger, S 1996, 'Formalising Design Exploration as Co-evolution: A Combined Gene Approach', *Advances in Formal Design Methods for CAD: Proceedings of the IFIP WG5.2 Workshop on Formal Design Methods for Computer-Aided Design*, Springer, pp. 3–30.

Shea, K, Aish, R and Gourtovaia, M 2005, 'Towards integrated performance-driven generative design tools', *Automation in Construction*, 14(2), pp. 253–264.

Shea, K and Cagan, J 1999, 'Languages and semantics of grammatical discrete structures', *Artificial Intelligence for Engineering Design Analysis and Manufacturing*, 13(04), pp.241–251.

Stiny, G 1980, 'Introduction to shape and shape grammars', *Environment and Planning*, 7(3), pp.343–351.