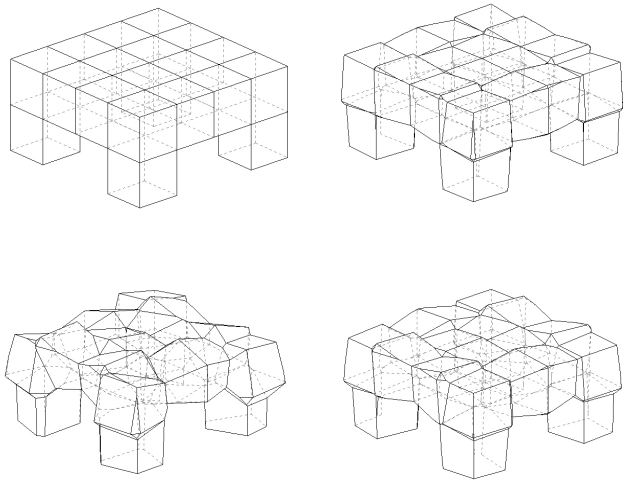
The Bartlett School of Graduate Studies - University College London

THE VORONOI DIAGRAM IN STRUCTURAL OPTIMISATION

EVA FRIEDRICH



This dissertation is submitted in partial fulfilment of the requirements for the degree of Master of Science in Adaptive Architecture & Computation from the University of London.

Bartlett School of Graduate Studies, University College London September 2008

ABSTRACT

In the course of the exploration of computational means in the architectural design process, in order to investigate more complex, adaptive geometries, the Voronoi diagram has recently gained some attention, as a structure that is modular but not repetitive, with potential for a great variety of complex geometries. The project looks at the Voronoi diagram as a load-bearing structure, whether its formal potential can be useful for statical optimisation, and how this optimisation can be facilitated. Hereby, the edges of the Voronoi polyhedra are regarded as structural members of a statical system, which then is assessed by structural analysis software.

The project explores different techniques to optimise the cell structure in regards to its structural properties. It is suggested that the main challenge to optimise the polyhedra structure lies in the inherent complex relationship between the Voronoi cell cores and the resulting cell geometry. Cell geometries emerge from combinatorial properties of the underlying set of points and so there is no means to manipulate the cells directly.

Results seem to indicate that the emergent Voronoi polyhedra structure, although difficult to control due to its complex behaviour, can produce a very specific type of structural order, which is distinctively beneficial for a statical structure. Certain polyhedra structures which emerge from the Voronoi system appear to exhibit ‘synergetic’ system behaviour of cooperating tear and pressure forces, which makes the structure more stable and minimises deformation. The overall shape of the structure hereby is minimally transformed. The working principle of the optimised structure lies rather in how forces are redirected in the system due to its specific topology, rather than through geometrical transformations of the structure’s overall shape.

Word count: 10.789

CONTENTS

FIGURES

ACKNOWLEDGEMENTS

With very many thanks to:

My supervisors

Christian Derix for conceptual advice which helped to channel my ideas, for the encouragement and support

Sean Hanna for his inspirational discussions, technical advice and encouragement

Alasdair Turner for the great, enjoyable and comprehensive course!

And also to:

Tristan Simmonds for his introduction to Oasys GSA, his advice on the analysis setup, and the very helpful reviews of the analysis results

Daniel Glaessl for architectural and conceptual discussions

1 INTRODUCTION

1.1 BACKGROUND

Alongside the introduction of computation in the design process, architects and structural engineers have been exploring the possibilities of more complex geometries and adaptive forms and structures. The Voronoi diagram has recently gained some attention in this field, being a three-dimensional space-filling structure which is modular but not repetitive. This makes it attractive as a design tool - as the non-repetitiveness suggest a parametric quality that might be more ‘adaptive’ than ordinary modular systems. A further interesting property of the Voronoi structure is its inherent notion of spatial relationships, adjacencies and neighbourhoods, of which the Voronoi cell is a direct geometric equivalence. This property suggests that the Voronoi diagram can be useful to model parametric spatial relationships and can thus find various application in architecture and urban design. Thirdly, architects and designers have been attracted by the ‘creative’ potential of the Voronoi diagram to produce emergent, complex cell geometries. Accordingly, the Voronoi diagram has been applied to experimental form finding tasks in various design projects.

However, the actual geometry of the Voronoi polyhedron is difficult to predict and control, as the shape of a cell emerges from the configuration of the entire neighbourhood of cell cores and is thus a result of multiple parameters. The geometry and the topology of the polyhedron of each cell – such as size, proportion or the number of edges – is highly sensitive to even the slightest change of position of any point in the neighbourhood. The relationship between the cell cores and the resulting geometry of the cells is thus inherently complex.

For architects and structural engineers, however, a tool which offers little control over the actual geometry obviously is of limited usability. - Being precise about the geometry of space and structure is what these professions are concerned with in the first place. So although the Voronoi diagram is likely to work well for tasks to optimising topologies, it remains unclear in how far the difficulty to control the cell shape is a limitation for its use as a design tool in architecture and structural engineering.

1.2 AIMS AND OBJECTIVES

Little research has so far been done to explore the Voronoi diagram under specific consideration of its geometrical properties rather than its cell topology. This project explores emerging geometries of three-dimensional Voronoi polyhedra in terms of their properties as load-bearing structures. This task obviously implies specific demands to the cell geometry as such. Similar to prior approaches, the motivation to become engaged with the Voronoi diagram is its potential for complex unexpected geometries, and it seems of interest whether this potential can be exploited to deliver optimised statical structures. However, it is unclear whether the Voronoi cell geometry can be controlled sufficiently to evolve statical structures at all.

1.3 INTRODUCTION TO THE PROBLEM

The statical structure that shall be subject to optimisation will be constituted by the edges of the Voronoi polyhedra, and be assessed by structural analysis tools in terms of its stability and its deformation. The evolution of the structure is induced through relocation of the Voronoi cell cores. The

resulting changes of the cell geometry are assessed statically and then feed back to guide the further evolution of the system.

Characteristically, when moving Voronoi cell cores, not only the Voronoi cell geometry changes, and hence the length and orientation of Voronoi cell edges, but also inseparably the actual cell topology. As neighbourhood relationship between cells changes, entire polyhedra faces emerge or vanish. Geometric constellations of the cell structure are created from very specific configurations of cell cores, and can equally vanish again through only slight changes of the configuration.

These topology changes, as will be shown, have a significant impact on the performance of the structure. The actual fitness of the overall structure will abruptly change at thresholds where Voronoi cell core movements cause the topology to change. In this sense, the changes of the Voronoi cell core configuration and the resulting changes of the overall structural fitness are non-linearly related. This ‘emergent’ property of the cell topology will make it difficult to predict successful future movements of Voronoi cell cores based on their former successes.

However, it will be suggested that there is also a significant potential: A great variety of different topologies can be created with slight changes of the Voronoi cell core configuration and thus minor changes of the overall shape of the structure as such. It shall be suggested that from this solution space of ‘near-by topologies’, solutions with significant structural potential can be retrieved: Topologies emerge which apparently redirect forces more efficiently than in the original simpler structures, whilst exhibiting ‘synergetically’ collaborating tear and pressure forces. Notably, these ‘synergetic’ structures show a comparatively good performance with only little actual deformation of the Voronoi cells themselves.

2 RELATED WORK

A considerable range of research has been done to investigate the potential of the Voronoi structure as a generative tool in architecture and urban design. Typically, the aim has been to generate functional topologies of entities, by letting Voronoi cells represent these entities, and to evolve the system's topology through optimising Voronoi cell neighbourhoods. The resulting cell geometry provides a structure, which accounts for these topological affordances in terms of its cell adjacencies. The cell structure can hence be considered as functional in terms of the topological affordances, or, at least, as a good starting point for further geometrical amendments to become functional. – However, little research has been done which targets the cell geometry as a main subject to optimisation, by defining specific fitness goals for the cell geometry itself in the first place.

2.1 THE VORONOI DIAGRAM AS A TOOL FOR FORM-FINDING

The Voronoi structure has been used as a tool for form finding, deliberately exploiting the 'creative' emergent potential from the geometry from bottom-up self-organising processes of the Voronoi point cloud. It is this formal potential that makes the Voronoi diagram attractive as a design tool.

M-any (Bonwetsch et al, 2006) is a research project from the ETH Zurich which explores the formal spectrum of emergent geometries of a parametrical Voronoi structure (Figure 1). The parameters which control the process are concerned with topological aspects of the cells in the first place. The geometry itself is hardly constrained with specific affordances but rather appreciated as emergently delivering unexpected results.

Formally, the resulting structures have been associated with foam-like structures such as sponges, bone structures and crystals. The tradition of these formal associations reaches back to the famous work of architects like Toyo Ito (Ito, 2008), Buckminster Fuller (in: McHale,1962) or Frei Otto (Otto, 2008), which was inspired by formation principles, geometries, spatial effect and constructions in nature, as a formal, spatial and/or constructive inspiration.

Figure 1: M-any: Parametric design based on the Voronoi diagram

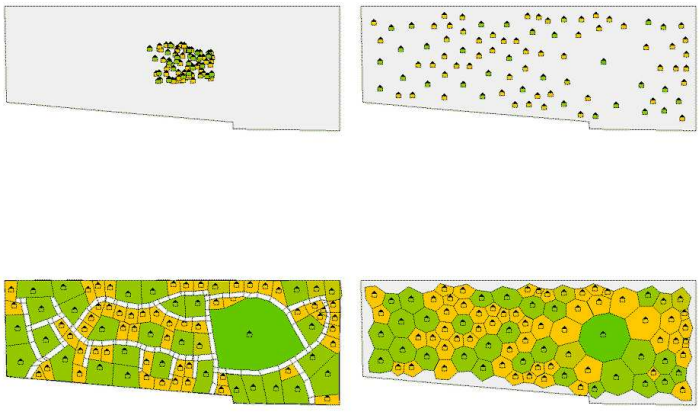
2.2 THE VORONOI DIAGRAM IN OPTIMISATION TASKS

The Voronoi diagram has been applied to spatial optimisation tasks in urban and architectonic contexts. An example which optimises topology in the first place, but though defines geometric affordances, is the research project Kaisersrot (Braach et al, 2006). In the course of this research project, a computer software was created which optimises land use for housing developments. The representation of the plots is based on the

Voronoi diagram. The tool generates layouts for housing developments, according to individual parameters that are individually defined for each plot, such as desired adjacencies, attractors and plot sizes. The optimisation process works in two stages, firstly, topological and layout parameters such as plot sizes, land use deviation and neighbourhoods of plots are evolved, secondly, the actual geometry is fixed in order to function with a street layout.

In the first stage, Voronoi cells - the plots - move around in order to increase their individual fitness, in terms of sizes and adjacencies. This eventually leads to an acceptable solution for the overall configuration topology. In the second stage, the actual cell geometry is corrected to function as an urban layout. Plot edges need to be aligned and straightened in order to allow of a reasonable street layout. These corrections are facilitated through alignments of the ‘plot’ cells with the adjacent ‘street’ cells (Figure 2).

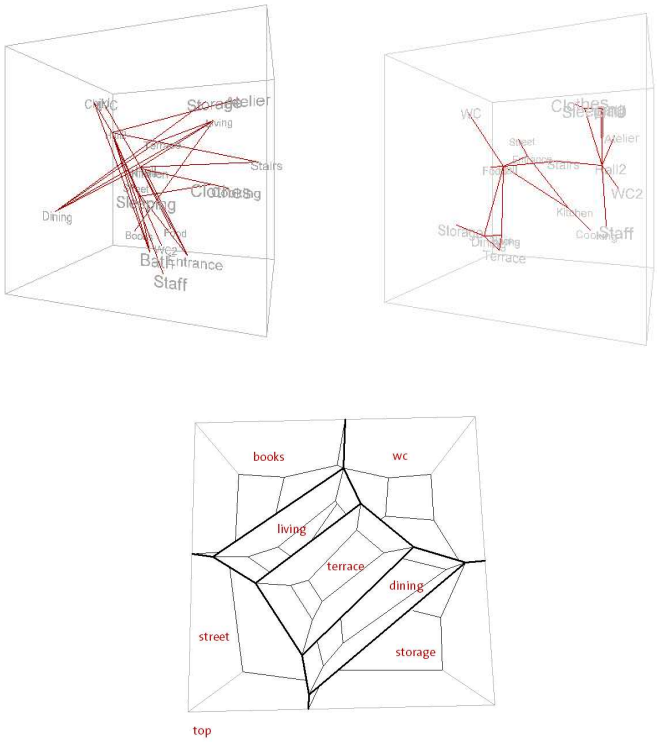
This example demonstrates that a Voronoi structure, which is optimised in terms of a desired topology of its cell layout, is not determined in terms of its geometry. There exists a great variety of geometries for a specific topology. Meaningful geometric order does not emerge from sorting out the cell topology alone.



Clockwise from top left: 1Initial configuration 2 Optimised configuration of plot cell cores 3 resulting plot layout from topology rules 4 plot layout after geometry optimisation

Figure 2: Kaisersrot. Urban layout generation based on the Voronoi diagram

In the project 'Self-organising Room Layout' (Friedrich, 2007) it was attempted to optimise the layout of 'rooms' in three-dimensional space. The rooms were given as point locations and their interconnections and desired adjacencies were represented as a graph of links. This graph of nodes and interconnections was optimised in three-dimensional space using a Kohonen algorithm. The Voronoi diagram was used to display the shape of the rooms, using the nodes as cell cores. Results show clearly that mere optimisation for adjacency is by far not sufficient to produce meaningful spatiality (Figure 3).



Clockwise from top left: 1Initial random configuration 2 Optimised configuration of room nodes through the Kohonen algorithm 3 Resulting room geometry as generated through the Voronoi diagram of the room nodes

Figure 3: Self- organising room layout using Kohonen Neural Network and Voronoi diagrams

2.3 THE VORONOI DIAGRAM IN STATICAL OPTIMISATION

Little research has so far been done considering the Voronoi diagram as a tool for structural optimisation of statical structures. Rather than the Voronoi cells, its dual, the Delaunay tessellation has been applied to structural problems (Canzarra, 2001), taking advantage of the fact that the three-dimensional Delaunay is a statically rigid structure of space-filling tetrahedra (Figure 4).

It remains unclear if the Voronoi system can be evolved for the specific geometric and structural affordances. The project presented here will investigate different optimisation strategies, aiming to evolve the structure of the Voronoi cells rather than the Delaunay tetrahedrisation in terms of its fitness as a statical system. The structural system to be assessed will hereby be generated from the Voronoi cell edges as structural beams, and will be evolved through feedback-driven amendment of the Voronoi cell cores.

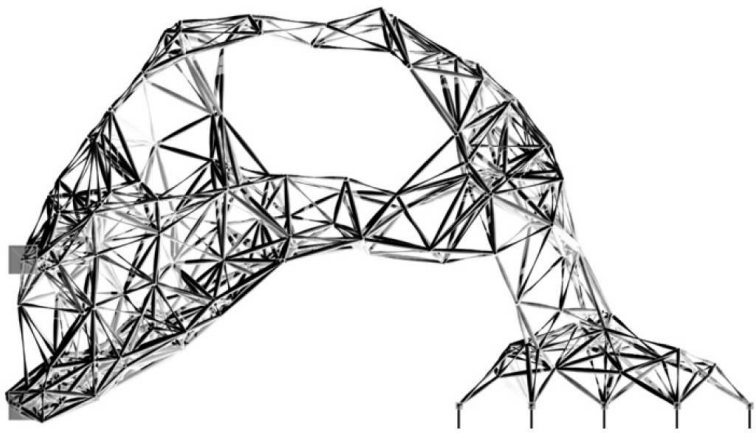


Figure 4: Pablo M. Canzarra – Evolved Cantilever

3 METHODOLOGY

3.1 COMPUTER TOOLS

The software which was developed for this project consists of two interacting components:

- a program written in Processing[1] which generates a statical structure based on a three-dimensional Voronoi diagram and optimises it through amending the Voronoi cell core configuration. In this component, different algorithms have been implemented which evaluate structural performance feedback during the optimisation process and amend the cell core configuration based on this feedback

- a structural analysis program - Oasys GSA[2] - to assess the structure and to feed back fitness results to the main program

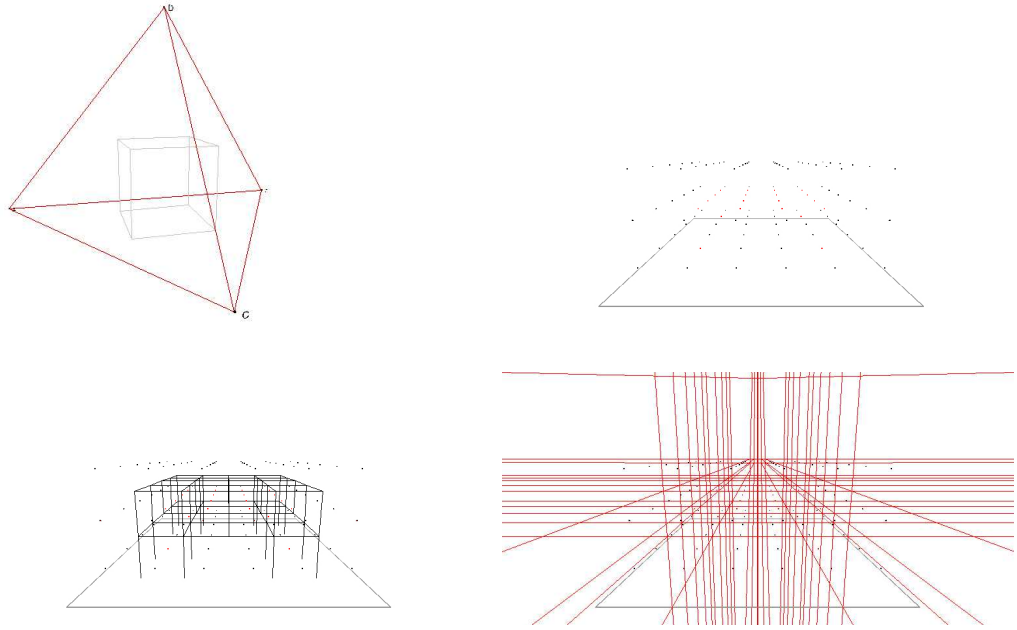
Oasys GSA can be controlled remotely via a COM-interface, so it can run in the background, and the process of analysing models and reimporting results can seamlessly be integrated into the Processing applet.

[1] www.processing .org

[2] www.oasys-software.com

3.2 THE VORONOI DIAGRAM AS A STATICAL STRUCTURE

The Voronoi diagram is created from a set of points inserted into a hyper-tetrahedron which confines the structure. The statical structure subject to optimisation is obtained from the cells of a subset of the point set (Figure 5). To confine the structure to the outer space, the subset of ‘structural’ cells are surrounded by a layer of ‘non-structural’ cells, which are assigned a constant position in space and are thus not actively taking part into the optimisation process.



Clockwise from top left: 1 Initial hypertetrahedron from four points 2 A set of 108 cell cores arranged on a grid 3 The Voronoi cell structure from the point configuration 4 The statical system from the inner layer of cells; cells are clipped on the bottom plane

Figure 5: The Voronoi structure as a statical system

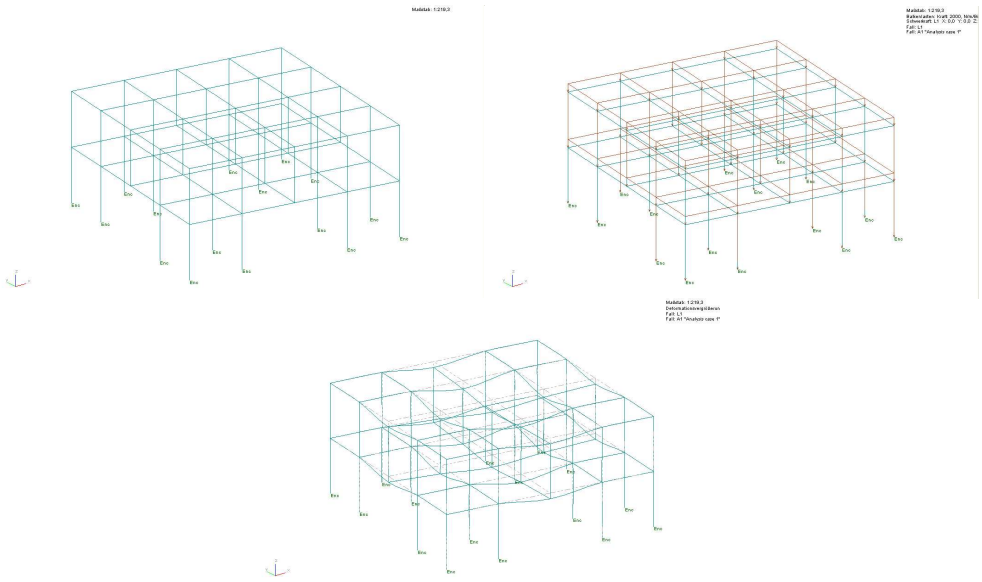
The edges of the structural cells act as beam members and are connected through rigid nodes – as the cell structure is not triangulated this is a necessary precondition to generate a valid statical structure which can be processed by the statical analysis software. In order to confine the structure towards the ground, the structural cells are clipped on a horizontal ground plane. Each beam member which intersects this plane is trimmed at the intersection point and fitted with a fixed support to the ground. Members below the bottom plane are not considered.

In order to analyse the structure in Oasys GSA, it has to be assigned some real-world properties such as size and material properties of the beam members. As the work presented here is more concerned with relative performance rather than to address a specific optimisation problem, it shall be suggested that these units can be chosen arbitrarily – within reason – as long as they are used consistently. Given that the size of the structure is in reasonable proportion to its beam sizes, GSA produces valid results, and thus gives meaningful account of relative variance of performance during the structure's evolution over time.

The optimisation process starts from a simple initial configuration: Voronoi cell cores are arranged on a grid, thus producing cube-shaped Voronoi cells. The resulting statical structures is a modular space frames with rectangular beam connections. The grid size has been set to 10x10x10 metres. At that scale, a reasonable material size for beams seems to be a circular hollow steel profile with a diameter of 0.3 m and a wall thickness of 0.02 m. Beams which connect to the bottom plane are defined as fixed supports.

In order to choose a simple load case, the structure is merely considered in terms of self weight. In a prior investigation (Friedrich et al, 2007), in some cases a moderate wind load has been applied which was implemented as

horizontal force of 1 kN/m^2, and suction on roof areas, as well as additional floor loading. However, it has been apparent that these load cases did not affect significantly the nature of the optimisation process. Hence, in the present work only self-weight shall be considered.



Clockwise from top left: 1 The structure as space frame, imported into Oasys GSA. 2 Load diagram 3 Deformation of the structure due to its loading (magnified x 12.5)

Figure 6: OASYS GSA – Statical analysis

The initial configuration consists of twenty structural cells, of which sixteen constitute an upper level which is supported at the ground level by four cells at its corners (Figure 6). Other configurations have been tested elsewhere (Friedrich et al, 2007), but, for conciseness, shall not be included in this investigation.

3.3 THE VORONOI CELL GEOMETRY AS AN OPTIMISATION PROBLEM

Oasys GSA calculates the values of forces and moments in beams and supports, as well as the deformation of the structure due to loading, firstly for the initial configuration and then after each time the Voronoi cell cores have been moved. The analysis results are reimported into the main applet and feed back into the optimisation process, in order to evaluate the next movements of cell cores. The performance of the initial configuration – the *maximum occurring deformation (MD)* – will be the measure and scale to appraise the increase of fitness – the reduction of this maximum deformation - during optimisation.

Although feedback on optimisation success can be obtained in each optimisation step, it turns out to be by no means straightforward to derive strategies for successful next movements from this feedback. This is, as shall be suggested, due to the inherent complex relationship between the configuration of the Voronoi cell cores and the properties of the Voronoi cell structure, in terms of the mere geometry of the cells geometrically but also due to the affordances of the structure as a statical system.

It shall be recalled that the Voronoi diagram is a topological data structure in the first place, which describes neighbourhoods of given points. The Voronoi cells are emerging from configurational properties of this point cloud. The Voronoi structure can hence be described from two different points of view, namely the microstatic view which considers properties of the set of points, and the macrostatic view, which is engaged with the geometry of the cells. It will be suggested that the relationship between the microstate and the macrostate is inherently complex, and that the specific type of complexity observed here will make it non-trivial to control the macrostate from the bottom up, and to generate geometry features which

are at all advantageous in the light of the optimisation task – the overall structure as a statical system.

4 THE VORONOI SYSTEM AND COMPLEXITY

4.1 TYPES OF COMPLEXITY

In science, the term complexity has been characterised in several different ways. In general, complexity is tied to the idea of a system, a set of parts which have relationships among each other, whereby the source of complexity is either the occurrence of numerous elements in the system or the occurrence of numerous relationships between the elements (Wikipedia).

Warren Weaver (Weaver, 1948) distinguishes between two types of complexity, namely ‘organised’ and ‘disorganised’ complexity. According to Weaver, disorganised complexity is a matter of a very large number of parts, in which individual interactions can be seen as largely random, but which total behaviour can be understood using probability and statistical means. These regularities can be exploited to relate the properties of the micro- and macrostate of a complex system, for example, as Sunny Y. Auyang suggests, by installing so-called inter-level theories (Auyang, 1998). Such inter-level theories define mathematical shortcuts between the micro- and the macrostate of a system which can be accomplished for example through probabilistic approximations on – both largely idealised – micro- and macrostates.

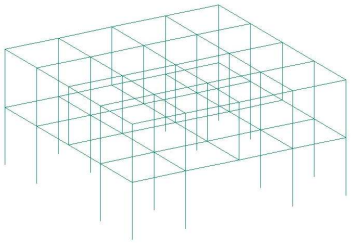
Different from disorganised complexity, organised complexity, according to Wikipedia, ‘resides in the non-random, or correlated, interaction between the parts of a system’ (Wikipedia). The system as a whole manifests, so Wikipedia, ‘macrostatic properties not carried by, or dictated by, these individual parts’. The organised macrostate emerges from specific configurations of its components - whereby the source of complexity in the

system lies in the combinatorial explosion of number of possible states with increasing numbers of entities in the system.

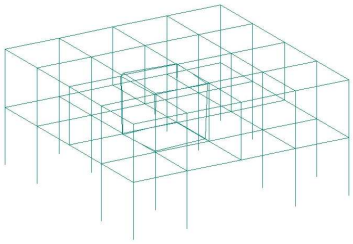
4.2 THE COMPLEX BEHAVIOUR OF THE VORONOI DIAGRAM

It shall be suggested that the kind of complexity of the relationship between the microstatic configurations of Voronoi cell cores to its macrostate, the resulting cell geometry, can be better understood in terms of the second definition of complexity, namely organised complexity, as the cell geometry emerges from specific configurational properties of the point cloud[3]. The respective geometrical features are a result of multiple local properties of the cell core configuration. For example, each plane between two neighbouring points depends on the relative position of those two points in terms of its orientation in space. To cut this plane to its actual shape as a face of a polyhedron we have to take into account the relation to all other planes to other neighbours which are intersecting this plane.

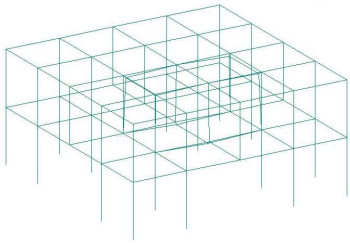
3 The Voronoi polyhedra structure defines topological neighbourhood of given sets of points by defining the space around each point which is closer to the point in question than to any other point in the point cloud. The dual of the polyhedra structure is the Delaunay triangulation, which joins neighbouring points through the vertices of its triangles. The Voronoi polyhedra are obtained from all vertical planes through the centre point of each of the Delaunay edges which connect neighbouring points. Each plane divides the space between two neighbouring points, defining the area which is nearer to each of them. Each polyhedron around a point has thus as many faces as there are neighbours to the point.



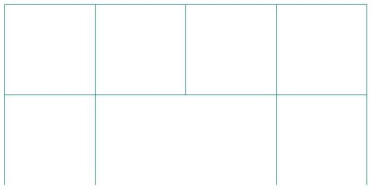
Voronoi Structure - Axonometry



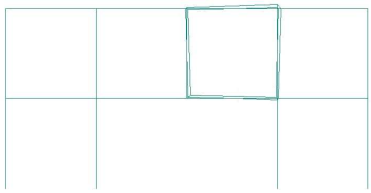
Voronoi Structure - Axonometry



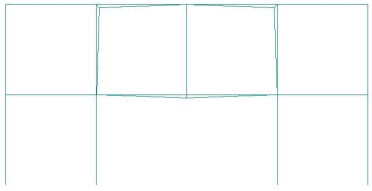
Voronoi Structure - Axonometry



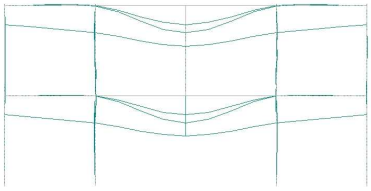
Voronoi Structure - Section



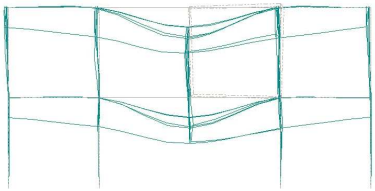
Voronoi Structure - Section



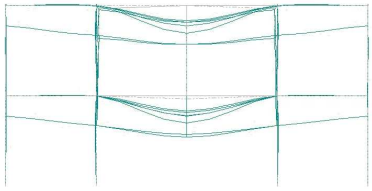
Voronoi Structure - Section



Deformed Image – Section
Deformation Magnification x50
Maximum Deformation = 89.62mm



Deformed Image – Section Deformation Magnification x50
Maximum Deformation = 102.1mm
Optimisation: - 13.9%



Deformed Image – Section
Deformation Magnification x50
Maximum Deformation = 82.44mm
Optimisation: + 7.8%

7.1 Voronoi Structure from a regular point grid

7.2 Voronoi Structure, one cell core is moved

7.3 Voronoi Structure, two cell cores are moved

The topology of the structure changes when individual cell cores are moved slightly out of the initial grid. In both cases (Figure 7.2 and 7.3) the geometry does not change much, however the topology substantially rearranges and becomes much more complex. Whilst the dislocation of a single point, as in (7.2), makes the structure less stable and increases deformation, in the second case, (7.3), the new topology shows more integrity – principal structural members stay connected to each other – and deformation is decreased.

Figure 7: Effects of cell core movements on topology and structural performance

When relative positions of Voronoi cell cores change, the resulting changes of the geometry and the topology of the cell structure are in no straightforward, proportionate relation to these changes: whilst the Voronoi cell cores are moving continuously in space, the impact on the macrostatic geometry can be anything from a linearly related stretching of the Voronoi cells, an angle turn of a face between two cell cores and an accordingly accelerated expansion or shrinking of some cell edges, or, on thresholds which lets a cell core lose one of its neighbours the vanishing of an entire face between two points, which causes the structure to substantially rearrange (Figure 7).

The source of complexity between the micro- and the macrostate of the Voronoi system lies thus on the one hand in the mere variety of possible cell geometries due to the exponentially increasing number of possible cell core configurations, and on the other hand in the non-linearly related effects of local microstatic changes to the global structure, on cell geometries as well as structural topologies simultaneously.

4.3 EXPECTED IMPACT ON THE OPTIMISATION PROCESS

It can reasonably be anticipated[4] that, in particular, the topological changes which are occurring when cell core configurations change will have a major impact on the fitness of the structure as a statical system. The specific configuration of structural members will matter for structural fitness in how forces can be transmitted more or less efficiently. The stability and integrity of the statical structure hereby will be crucially dependent on the connectivity of its members and the underlying topological and geometrical principles of the structure as a whole. If principal members lose connections, the stability of the structure will decrease (Figure 7). The Voronoi macrostate, as a statical structure, will thus be a self-contained system with distinct properties and affordances on this macrolevel which

[4]The architect Buckminster Fuller, for example, who has been extensively researching the property of configurations in statical structures, has pointed out that load-bearing structures must be considered as systems which cannot be analysed by considering their individual parts in terms of strength and sum up their performance, but to take into account the intrinsic system behaviour of the structure as a whole. Buckminster Fuller coined the term 'synergy' when he referred to the specific system behaviour structural systems exhibit which is more than the sum if its parts. A synergetic structure Fuller developed is the Octet truss (Fuller, 1956), a space frame structure from alternating octahedra and tetrahedra, now widely used as wide-spanning joists. Fuller could demonstrate that 'the strength of the framework is far greater than would be predictable using any conventional formulae based on resolution of forces and known values of strength of materials. He writes that 'In fact, my practical tests have shown that the actual strength of these … structures so far exceeds calculated values that such structures are 'synergetic' in the sense that we have a stress behaviour in the system which is unpredicted by its parts.' (in: Meller, 1970)

are non-linearly related to properties of the microstate: structural performance will be dependent on the macrostatic system behaviour, the integrity of the structure as a whole. This structural integrity might be difficult to control from the bottom-up, through movements of the Voronoi cell cores.

In the optimisation task at hand, the system's *state space* shall be defined as the description of possible configurations of cell cores – each cell core configuration having a unique cell geometry associated on the macrolevel, and the *fitness landscape* shall be defined as the landscape of associated structural fitness levels of the Voronoi cell structure at each respective state. From above considerations it can be anticipated that a 'statical' Voronoi system travelling through its state space will encounter sudden topology changes at thresholds where neighbourhood relations among the Voronoi cell cores change, and that these topology changes have a major impact on the structural fitness. We thus can expect the fitness landscape of the statical system to be irregular and disrupt, with considerable slopes on thresholds where the topology of the structure changes.

5 OPTIMISATION STRATEGIES

5.1 OVERVIEW ON THE APPLIED TECHNIQUES

This paper will present two different optimisation strategies: an algorithm related to a gradient descent algorithm on the one hand, and on the other hand a strategy which searches exhaustively through combinatorial movements of Voronoi cell cores.

The first approach attempts to explore the gradient of the fitness landscape of the Voronoi system at the system's current state towards its neighbouring states. The technique presented here makes a guess on the gradient of the fitness landscape at the current state of the system by moving the individual cell cores one step in each of the three dimensions x, y and z and, from these test steps, evaluating a vector for a combined movement of all cell cores towards the system's next state, taking the achieved degree of increase in fitness as a weight for the respective dimension.

The second approach acknowledges the fact that geometric features of the Voronoi cells are results of specific configuration of the cell's neighbourhood rather than a statistical distribution of local properties. In each optimisation time step, possible next states are enumerated through searching exhaustively through possible combinations of cell core movements. Due to the immense size of the number of all possible combinations of movements for all Voronoi cell cores of the given system, however, a full exhaustive search is not feasible. Thus, exhaustive search is performed on subsets of the system – groups of points are moved sequentially.

Results of all series will be compared in terms of the respective degree of fitness that can be achieved as well as the specific geometric features which emerge from each technique. All techniques, over time, are able to produce structures which are optimised terms of the defined target to minimise deformation. However, there are considerable differences in the number of steps for each strategy to reach a certain degree of fitness on the one hand, and the degree of deformation of the cell geometry which comes with it.

It appears that finding beneficial topologies seems in fact to play a crucial role for optimisation success. Whilst the gradient descent – style approach - which tries to *approximate* each next advantageous state through evaluating isolated movements of Voronoi cell cores - seems to be less efficient to find suitable topologies, and rather is trading geometric deformation against topological efficiency, the approach evaluating local combinatorial moves seems in some cases to be able to detect topologies which improve stability with comparably little actual geometric change in cell shape.

The next sections will present the results of two gradient descent applications with different step sizes, and several series of combinatorial-movement optimisations, varying in structure of subgroups and sequence of processing these groups.

5.2 TECHNIQUE 1: ESTIMATING THE GRADIENT OF THE FITNESS LANDSCAPE

5.2.1 SETUP

The standard gradient descent algorithm (Wikipedia) aims to find a local minimum of a function, by calculating the gradient of the function at the current point, and then taking steps proportional to the negative of the gradient. In each iteration, the next state x_{n+1} is calculated through

$$x_{n+1} = x_n - \gamma_n \nabla F(x_n) \qquad [01]$$

with $\nabla F(x_n)$ being the gradient of the function at x_n, and γ_n being the step size.

Given a system which is constructed from the Voronoi cells of 20 Voronoi cell cores, the state space of the configurations of the 20 cell cores can be described using a vector of $20 * 3$ dimensions to hold the x, y and z coordinates of each of the Voronoi cell cores. The fitness of each state is the resulting performance of the statical structure of the Voronoi cells.

At each iteration of the optimisation process, a guess is made on the gradient of the fitness landscape of the system state space at its current state, in order to move the system into this direction. For this guess, each cell core performs a ‘test step’ in each dimension x, y and z, and the resulting difference in fitness compared to the current state is assessed. A high decrease in fitness for the respective test step indicates a steep gradient in the direction the test step was done. From these test

movements, a $20*3$-dimensional vector is generated which points into the assumed direction of the steepest ascent of fitness for the system at this state. This Vector is calculated by taking the direction of each test step, weighted by the amount of change in fitness of the test step - this should take into account the amount of increase or decrease in fitness each individual move has in relation to the other moves, thus test steps which have a greater impact will bias the final combinatorial move stronger than those with minor impact. This vector is applied to the x, y and z coordinates of the Voronoi cell cores in order to move the system to the next state.

$$\begin{pmatrix} x_{1_{n+1}} \\ x_{2_{n+1}} \\ x_{3_{n+1}} \\ \vdots \\ x_{20*3_{n+1}} \end{pmatrix} = \begin{pmatrix} x_{1_n} \\ x_{2n} \\ x_{3_n} \\ \vdots \\ x_{20*3_n} \end{pmatrix} + \gamma \begin{pmatrix} MD_n - MD_{(x_{1n}+1)} \\ MD_n - MD_{(x_{2n}+1)} \\ MD_n - MD_{(x_{3_n}+1)} \\ \vdots \\ MD_n - MD_{(x_{20*3_n}+1)} \end{pmatrix} \qquad [02]$$

The algorithm hence ‘guesses’ on the most advantageous direction towards the fitness optimum, by merging the test vectors of the individual moves due to their probability pointing towards the optimum of the fitness landscape. *Formula [02]* presents the $20*3$-dimensional vector which suggests the next state for the Voronoi cell cores, with $x_{1_{n+1}} \ldots x_{20*3_{n+1}}$ being the x, y and z coordinates for the 20 Voronoi cell cores at this stage, MD_n being the current maximum deformation and $MD_{(x_{1_n}+1)}$ to $MD_{(x_{20*3_n}+1)}$ the maximum deformation of the structure resulting from the test step of the respective cell core in the respective dimension. It is assumes that through

these guesses the system will eventually move to an overall better stage, and errors, fuzziness and false estimations will average out.

In below elaborations, to each sequence of performing all test steps of all Voronoi cell cores and the resulting step of the system to the next state will be referred to as one *iteration*, or one *optimisation step*.

In each optimisation procedure, a constant step scale γ has been used throughout. Several different scale values have been tested in different experimental series. The series using step scales $\gamma = 0.01$ and $\gamma = 0.001$, the latter being a rather fine grained step size, shall be presented in this paper. Below, these two optimisation procedures shall be referred to as $[gd0.01]$ and $[gd0.001]$

5.2.2 RESULTS

In both series, the system got better over time in terms of the targeted fitness goal of minimising the maximum nodal deformation (Figure 8, Figure 9). However, there are substantial differences between the series in terms of the characteristics of the optimisation process as such, the number of iterations which were needed to reach a certain level of optimisation, the maximum achievable optimisation and the properties of the emergent cell geometry.

Both series have been running for about 500 iterations. $[gd0.01]$ optimises rather quickly in the very first steps, and minimises deformation by 40%, compared to the deflection of the initial structure, then gradually slows down. After 150 iterations, the system did not appear to optimise further, but contrariwise, tended to 'mess up' again. $[gd0.001]$ optimised very slowly, but steadily over a very long time, and seemed to be able to optimise even

further after 500 iterations if it would have been running on. However, it needed about 100 iterations to achieve a comparable value than $[gd0.01]$ already reaches in the first couple of iterations.

In terms of totally achieved performance, again $[gd0.01]$ seemed to perform better: It accomplished about 60% optimisation after 160 time steps. $[gd0.001]$ reaches about 50% optimisation after 500 iterations – in a slow but rather steady process.

Comparing the geometry of the optimised structures, $[gd0.01]$ produces stronger deformed geometries of cells than $[gd0.001]$. However, the geometry looks very irregular and to some extend even random - which appears inadequate given this very regular symmetrical structure and the trivial load case – any optimisation of the shape would have been expected to reflect these regularities. Moreover, there is evidence that in some cases, that assumptions on the gradient of the fitness landscape were misleading: For example, in the first iteration of $[gd0.01]$, all four Voronoi cell cores in the centre of the upper layer have been moved in the same direction– namely in the opposite direction than their initial test steps – which indicates that each test step must have made the system worse at this stage, and the program now just moved the cell cores in the opposite direction of the test steps.

$[gd0.001]$ generally made more subtle changes to the geometry due to the smaller step size, which produces cells which are not strongly deformed, but rather contorted and shifted against each other, whilst maintaining the original sizes and proportions. However, the geometry appears to be quite irregular nevertheless.

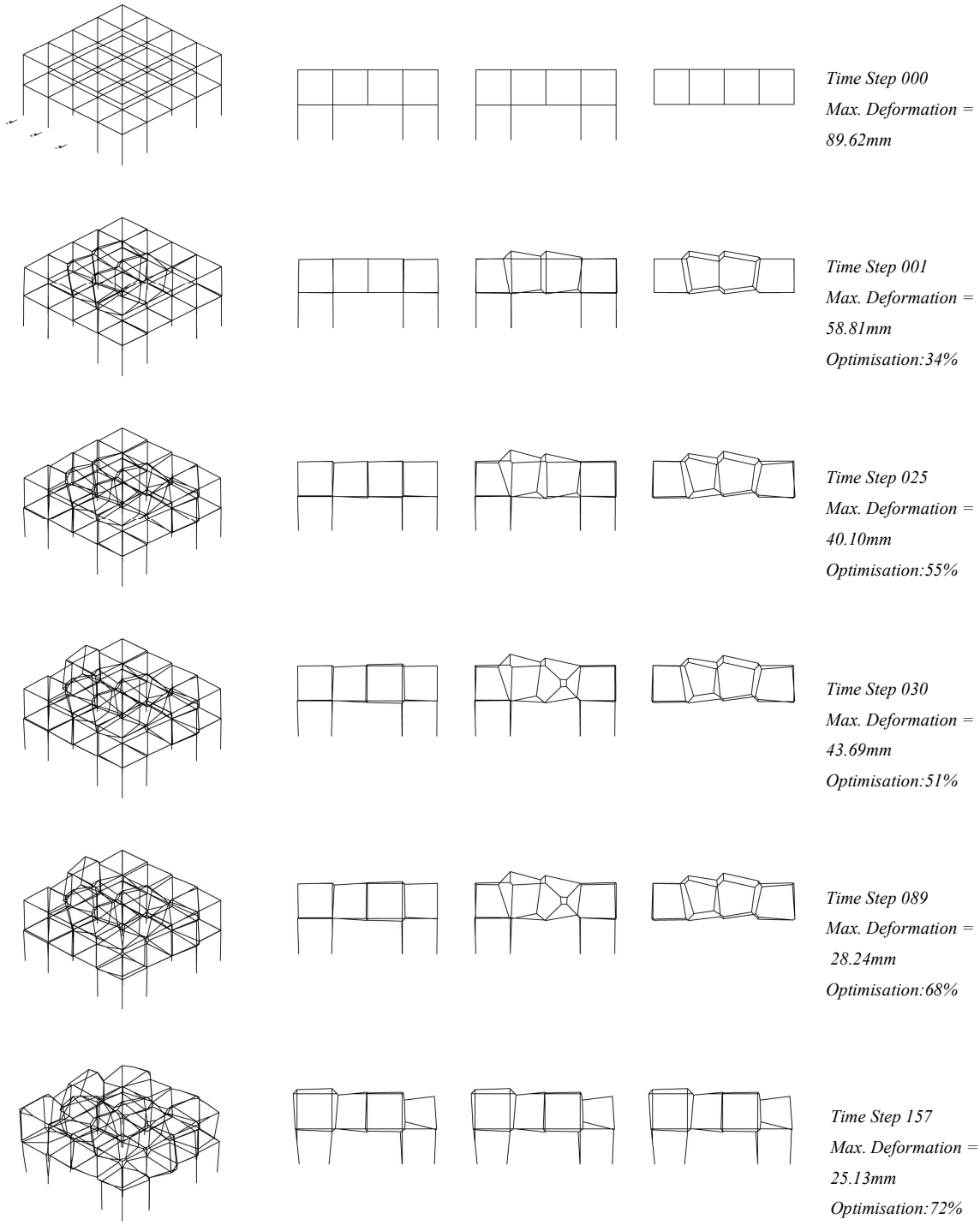


Figure 8: Evolution of $[gd0.01]$ ***over time***

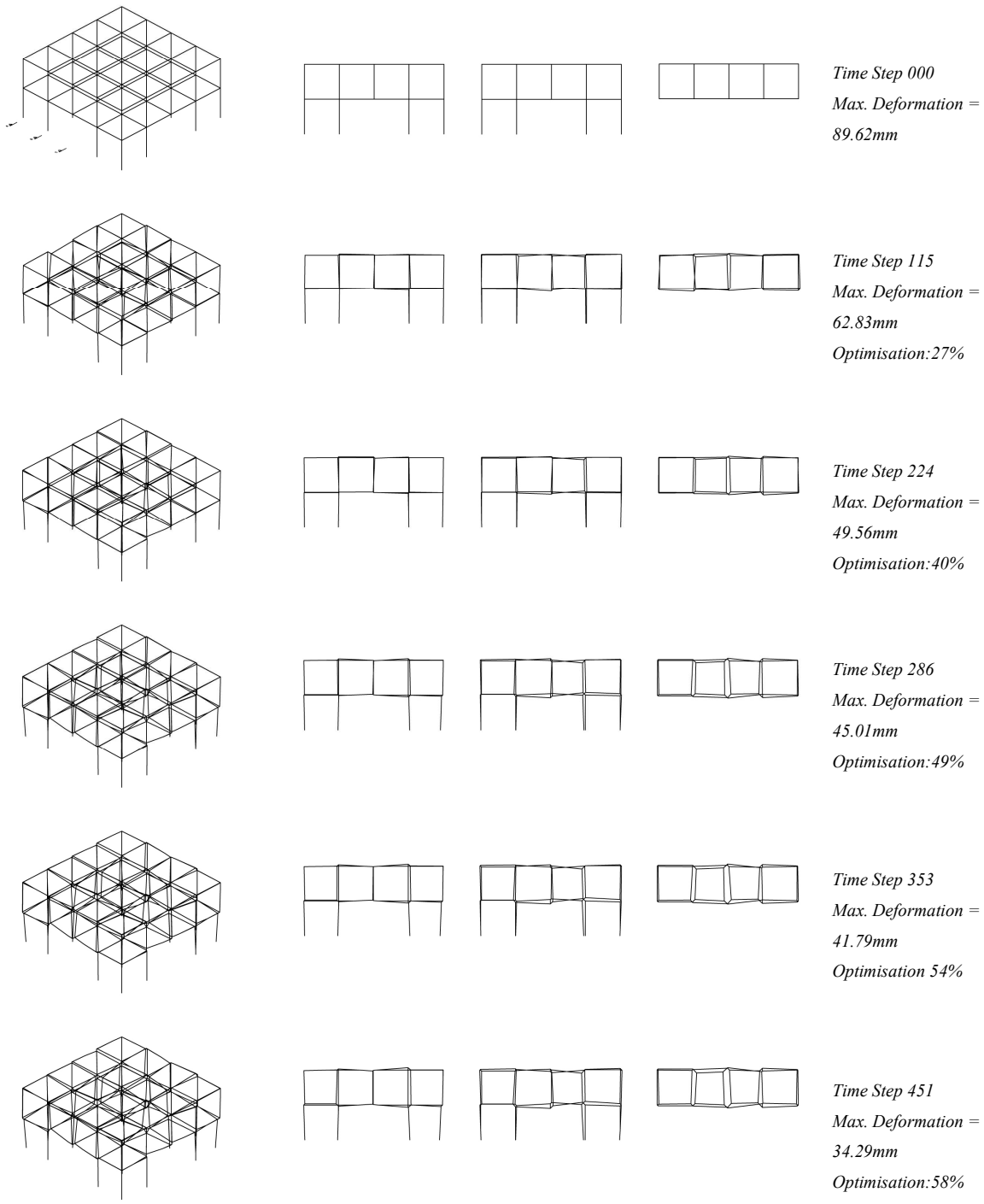
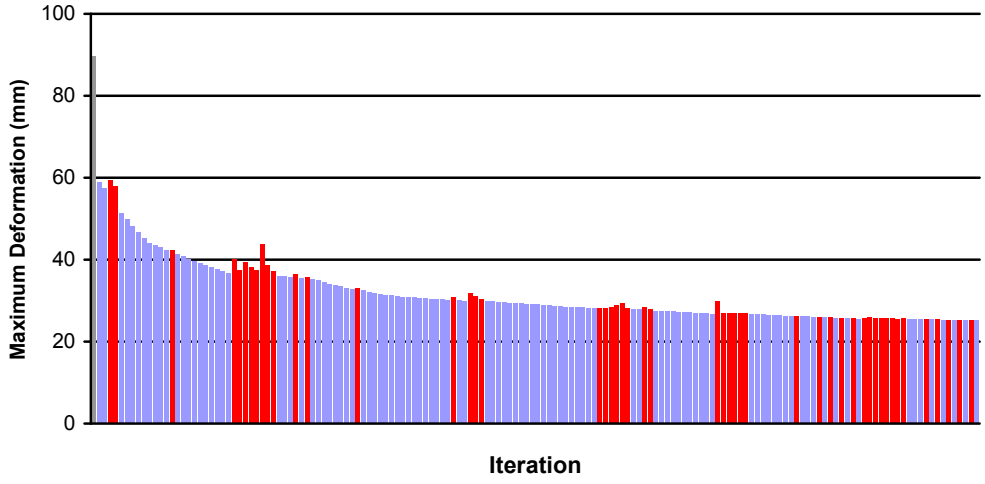
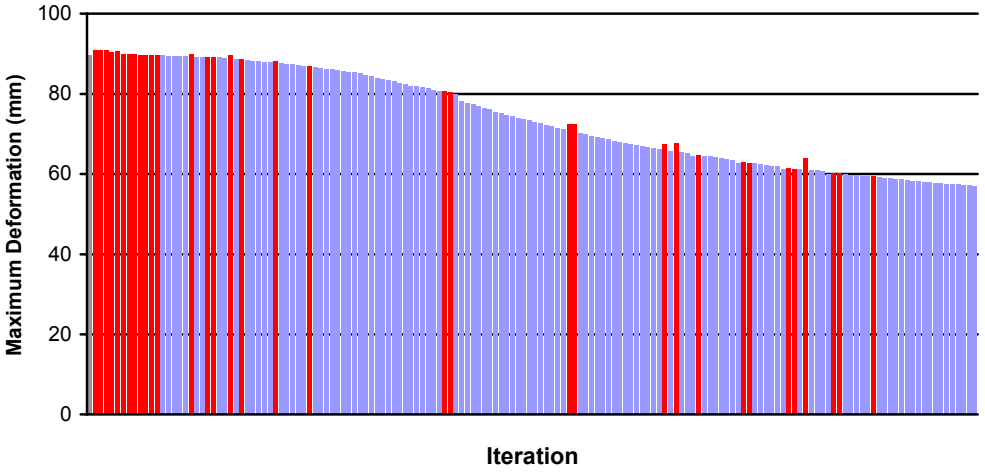


Figure 9: Evolution of $[gd 0.001]$ ***over time***



The chart shows the maximum deformation for the first 160 iterations of the optimisation process. The first bar (grey) represents the original structure. Bars in red indicate iterations in which the system got worse.

Chart 1: Maximum Deformation of $[gd0.01]$ ***during optimisation***



The chart shows the maximum deformation for the first 160 iterations of the optimisation process. The first bar (grey) represents the original structure. Bars in red indicate iterations in which the system got worse.

Chart 2: Maximum Deformation of $[gd0.001]$ ***during optimisation***

One interesting finding, however, is that, comparing results of the two processes at stages of similar degree of optimisation, for example at deformation reduction of 50% - which happens at iteration 3 for $[gd0.01]$ and at iteration 400 for $[gd0.001]$ – actually the latter has achieved this performance with a substantially lesser amount of geometric change than $[gd0.01]$, but rather through topological alterations of the structure. Apparently, $[gd0.01]$ has 'overstepped' some local optima, and there are possible solutions with equivalent performance, accomplished through certain topologies which require a lesser amount of geometrical deformation.

5.2.3 PERFORMANCE ASSESSMENT

To summarise, it can be ascertained that with the technique described above a certain degree of structural optimisation can be accomplished. However, resulting geometries seem counter-intuitively irregular, and the optimisation process either seems to overstep local optima, as in $[gd0.01]$, or is progressing impracticably slow, as in $[gd0.001]$.

One factor which is constraining the algorithm might be the fact that the fitness landscape of the microstatic state space is inherently irregular and hilly, with jumps in structural performance due to ever-changing topologies of the overall cell members as Voronoi cell cores move around. In this case, the local gradient of the fitness landscape would be of limited significance to direct a system to a relative optimum. - In fact, it is an accredited weaknesses of the gradient descent algorithm that it 'can take many iterations to converge towards a local minimum, if the curvature [of the search space] in different directions is very different.' (Wikipedia)

There is furthermore reason to assume that structural features of the Voronoi macrostate – the cell geometry - can probably not efficiently be optimised by simply merging individual moves of single cell cores, weighted by their isolated success, to a combined step. This might be due to the kind of complexity that characterises the relationship between Voronoi cell cores and the respective cell geometry, namely organised complexity, which arises out of the organised complex interaction of Voronoi cell cores which produces the overall cell geometry. Each cell cores configuration produces a specific macrostatic cell geometry. There is an abundant number of possible macrostates which can significantly differ in topology and thus in terms of their fitness as a structural system. The fitness of the system therefore depends on the *specific* configuration of its microstatic entities rather than averaging out from a certain statistical distribution of local properties – as it can be observed in some large many-body systems. Optimising the cell structure of the Voronoi diagram thus might present a problem which resists efficient exploration through probabilistic means.

For example, referring back to (Figure 7), if we start from a regular configuration of cell cores which produce a regular modular space frame, and slightly move one cell core out of the grid, the system immediately undergoes major topological changes: principal members get disconnected, and the structure becomes less stable. Moving two critical cell cores in respect to each other however, can lead to a configuration in which topology changes lead to coherent structures beyond the individual cell – principal members keep in touch with each other. This complex geometry emerges precisely form a specific configuration of neighbouring points, and cannot be deduced from individual movements.

Stephen Wolfram has coined the term ‚computational irreducibility' for systems of organised complexity which resist mathematical approximation so that in effect the only way to find their behaviour is to trace each of their

steps” (Wolfram, 2002). If the Voronoi system presents a system which allows no mathematical shortcut between its microstate and the structural behaviour of its macrostate, one would need to explore the system by iterating through all its possible states.

In the next section, an optimisation strategy which systematically explores ranges of possible states is applied. It will be suggested that despite obvious drawbacks of exhaustive search it though provides interesting results in terms of geometry and topology of the Voronoi cells. It appears that, although only a limited range of the solution space can be considered due to its abundant size, exhaustive search gives more control over the macrostatic geometry and thus seems to drive forward the optimisation process more straightforwardly than the strategy presented before.

5.3 TECHNIQUE 2: EXPLORATION OF COMBINATORIAL CELL CORE MOVEMENTS

5.3.1 SETUP

Optimisation strategies which explore a solution space by enumerating all possible states are referred to as exhaustive search (Weisstein, 2008). Exhaustive search is usually applied to problems in which no efficient solution is known. The advantage of this technique is that a solution will always be found if one exists. However, as the computational cost is proportional to the number of candidate solutions, exhaustive search is usually limited to applications with limited problem size, or when there are problem-specific heuristics which can be used to reduce the set of candidate solutions to a manageable size (Wikipedia).

Searching exhaustively through a range of configurations of Voronoi cell cores and their respective geometries will allow to assess a variety of emerging topologies more systematically. However, there are two obvious drawbacks of the application of the technique on the optimisation task discussed here:

- Exhaustive search is usually applied on discrete problems. It is hence necessary to 'discretise' the possible movement for each cell core, to obtain a set of possible next positions to iterate through.

- The range of configurations which can be considered giving limitations of computing power will present a very small subset of all possible states.

To address the first issue, possible movement for each cell core has been limited to an array of movement vectors of the size 3^3 to allow for all combinations of x , y and z - movements being either -1 , 0 or $+1$. To scale movement to the system size, each movement vector has been scaled to the length of 5 cm.

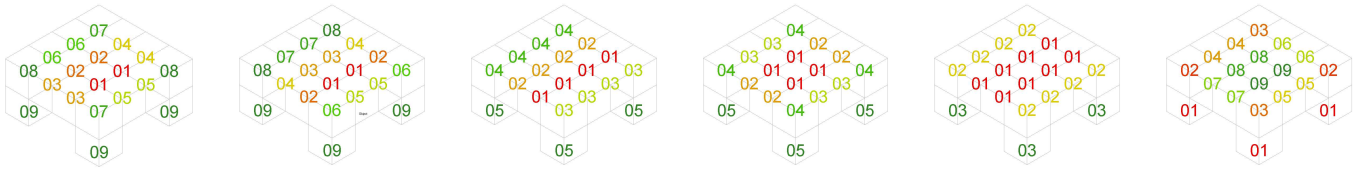
A full exhaustive search optimisation would now, in each loop, enumerate all possible states from all combinations of movements for the set of cell cores, assess the structural fitness of the resulting Voronoi cell structure and finally execute the combinatorial move which has shown to be the best option.

With 3^3 possible next positions for each Voronoi cell core, the number of possible next states in a system of 20 cell cores in each cycle is $3^{3^{20}} = 42.391.158.275.216.203.514.294.433.201$. It is thus not feasible to enumerate this number of possible states, as processing time for one analysis loop takes about 0.5 seconds on a standard laptop when using the

software components as described above. Thus, it would not even be possible to search all possible states of only one loop of the optimisation process.

In order to limit the size of the search space, the whole set of Voronoi cell cores is hence divided into subsets. The optimisation process enumerates the possible system states from movements of members of one group at a time, chooses the best option and moves on to the next group.

In the work presented here, six different experimental setups, which differ in their constellations of subgroups, shall be compared, in the following referred to as $[cm01]$, $[cm02]$, $[cm03]$, $[cm04]$, $[cm05]$ and $[cm06]$ - see Figure 10.



$[cm01]$	$[cm02]$	$[cm03]$	$[cm04]$	$[cm05]$	$[cm06]$
- groups of 2	*- groups of 2*	*- groups of 4*	*- groups of 4*	*- groups of 8*	*- groups of 2*
- starting from the centre of the structure	*- processing cells in rows*	*- processing cells in rows*	*- starting from the centre of the structure*	*- starting from the centre of the structure*	*- reverse order than [cm01]*

Figure 10: Six experimental setups of cell core groups to be moved in respect to each other during optimisation

It appears to be reasonable to group together adjacent cell cores in order to support potentially emerging larger-scale structures between adjacent cells; and to group together cell cores in similar positions in the structure, in order to apply a successful movement to multiple similar members in one go. Subsets of each two, four and eight cell cores have been tested.

The number of candidate solutions to be assessed each time is 729 for subsets of 2 cell cores, 531.441 for subsets of 4 cell cores, and 282.429.536.481 for subsets for 8 cell cores. This still leaves us with a considerable number of combinations per subgroup. From these, only a small percentage of combinations can be anticipated to produce structures which present an enhancement in fitness. In fact, test runs which enumerated all possible solutions for subsystems of pairs of two cell cores have shown that there is a high agglomeration of advantageous solutions among the combinatorial moves which are done symmetrical to each other, for example the two Vectors

$$[\upsilon_1] = \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix}$$

$$[\upsilon_2] = \begin{bmatrix} 0 \\ -1 \\ 1 \end{bmatrix}$$

for two cell cores moving symmetrical to a plane in y – direction (Figure 11). Typically, these symmetrical movements tend to produce cells which major cell edges remain connected to those of the adjacent cell, thus maintaining straightforwardly connected topologies beyond cell borders. Most other combinations of moves will rather create a greater disruption

and topological discontinuity between adjacent cells, which tends to decrease structural stability of the overall structure.

In order to shrink down further the set of candidate solutions to be searched through, it therefore seems reasonable to only apply symmetric movements, according to a symmetry plane through the centroid of each group. Thus, for each group – equally for two, four and eight Voronoi cell cores - only 27 solutions need to be tested (Figure 12).

Adopting these presettings, the optimisation process is carried out by processing cell core groups sequentially: the solutions of possible symmetrical movements of members of one group at a time are enumerated, the best option is selected, then the next group of cell cores is processed. The operation of processing all groups once shall be referred to as one *iteration*, or one optimisation step.

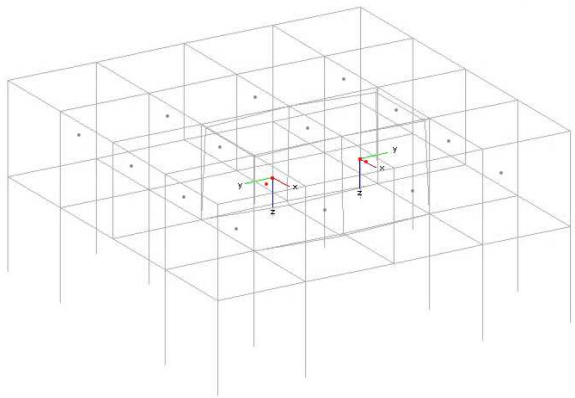
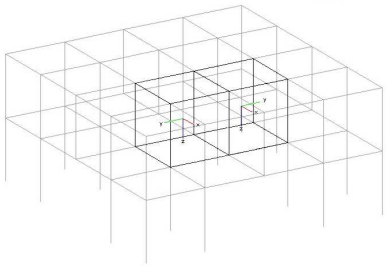
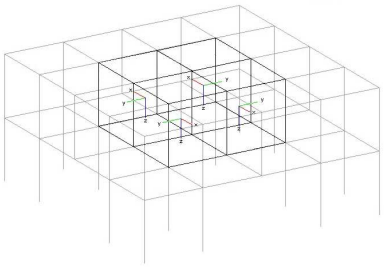


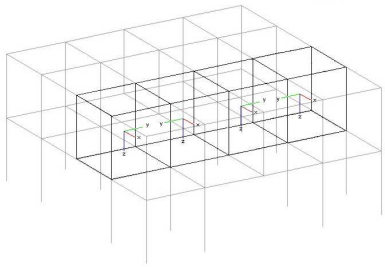
Figure 11: Two cell cores moving symmetrically to each other on local mirrored coordinate systems



Group of 2 points
Symmetry along the vertical plane through the group centroid



Group of four points
Symmetry axis is the vertical axis through the centroid of the configuration



Row of four points
Each two adjacent cells will behave symmetrical to each other

Points will be moved symmetrically to each other according to their local coordinate systems. These will be mirrored according to the group centroid.

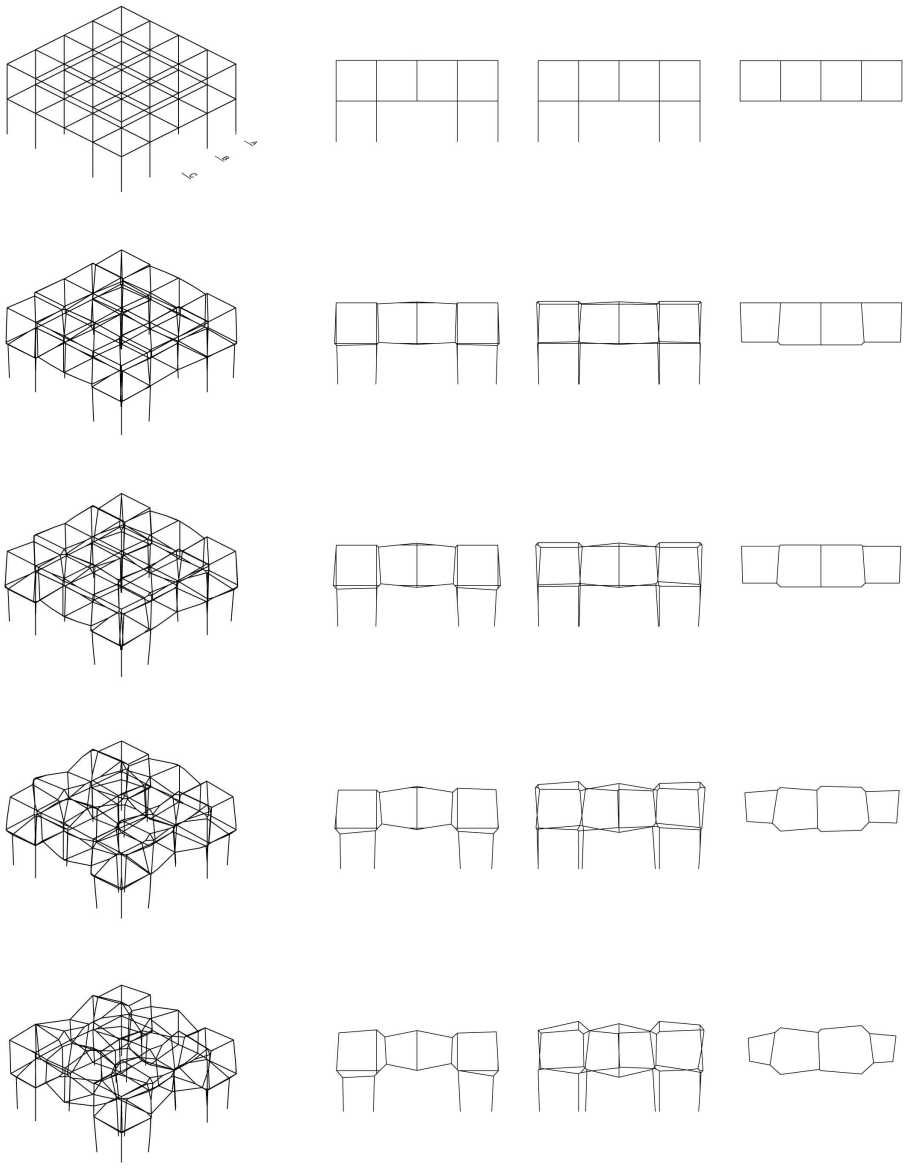
Figure 12: Combinations of movements of point groups

5.3.2 RESULTS

Figure 13 shows the evolution of $[cm01]$ over time. In terms of the optimisation target to minimise the maximum nodal deformation of the structure, an optimisation of 86.4% is accomplished after 17 iterations. The major part of the improvement hereby is facilitated in the early stages of the process. In the later stages, improvement rates significantly slow down (Chart 3)

The development of the Voronoi cell geometry shows a sequence of characteristic formal stages over time: In the early stages, it is rather the topology of the structural members which evolves and becomes more complex, whilst the Voronoi cells themselves do not change very much but just slightly contort against each other. Cell edges are rotated out of the previously orthogonal arrangement and appear to create braced configurations which operate on a larger scale throughout the structure.

Typical topological features which are established in these early steps of the process tend to persist in later stages. During the later stages, the geometry of the structure develops more significantly. The cells are no longer cube-shaped, but become more complex and multi-facetted. The overall structures now resemble foam-like formations. The structures also tend to get more irregular and asymmetrical. However, the impact on the actual fitness of the structure is rather small compared to the degree of geometrical change the structure undergoes.



Time Step 000
Max. Deformation = 89.62 mm

Time Step 001
Max. Deformation = 45.82 mm
Optimisation:48%

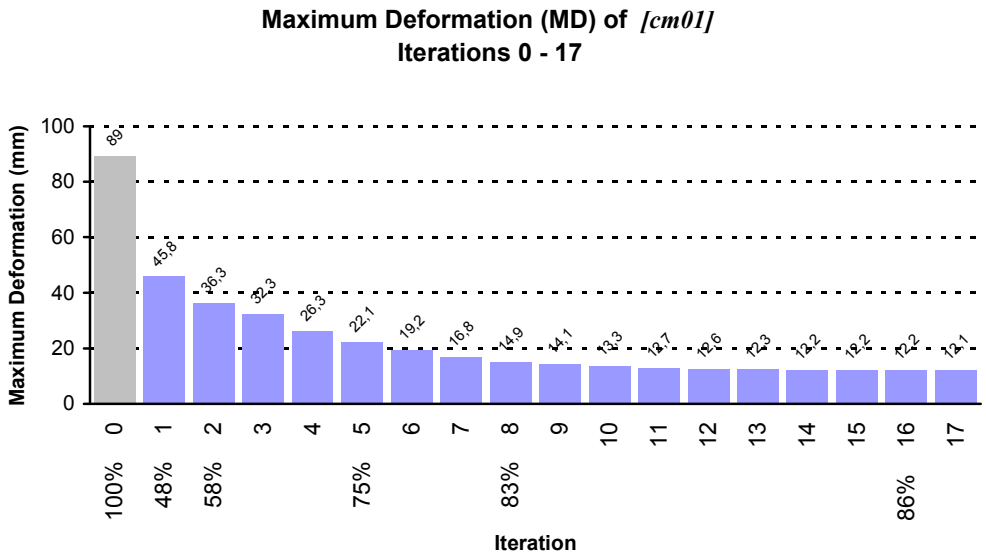
Time Step 005
Max. Deformation = 22.15 mm
Optimisation:75%

Time Step 008
Max. Deformation = 15.09 mm
Optimisation:83%

Time Step 016
Max. Deformation = 12.2 mm
Optimisation 86%

Axonometry *Section A-A* *Section B-B* *Section C-C*

Figure 13: Evolution of $[cm01]$ ***over time***



The chart shows the maximum deformation for the first 17 iterations of the optimisation process.

Chart 3: Maximum Deformation of [cm01] ***during optimisation***

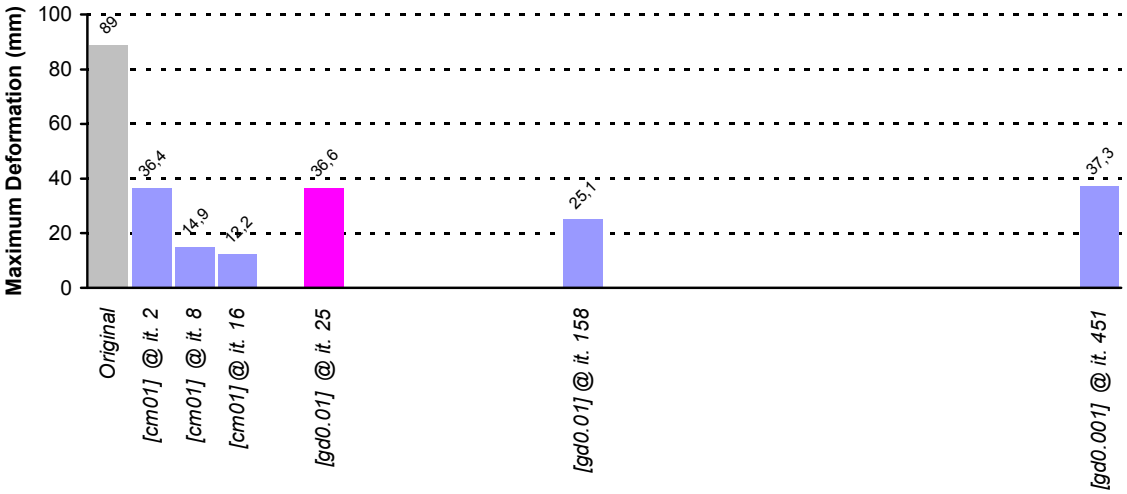
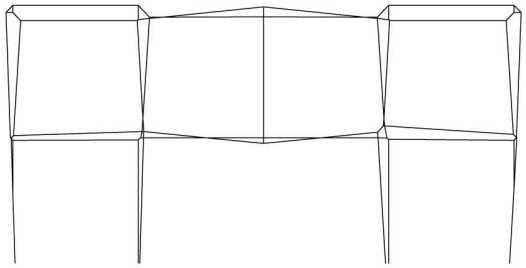
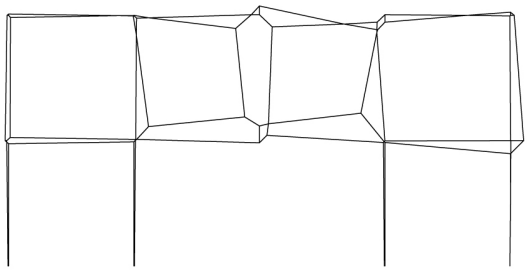


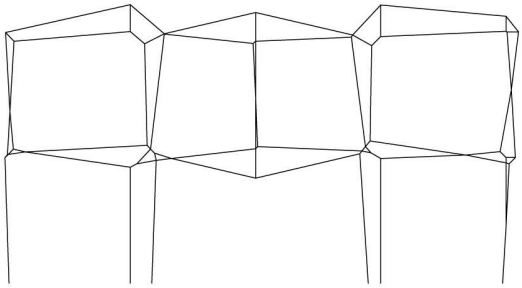
Chart 4: Maximum Deformation (MD) of [cm01], [gd0.01] & [gd0.001] at different stages of optimisation



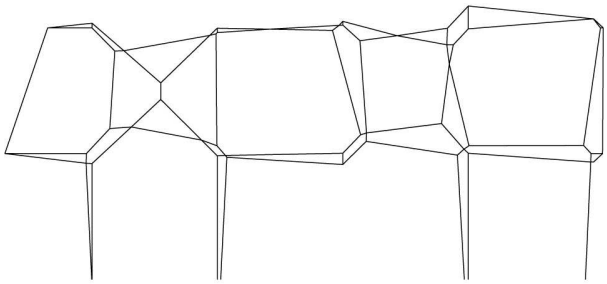
15 - 1



14 – 1



15 – 2

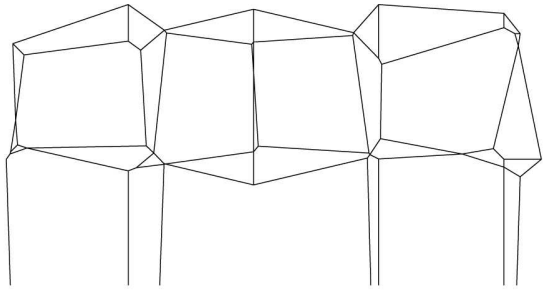


14 - 2

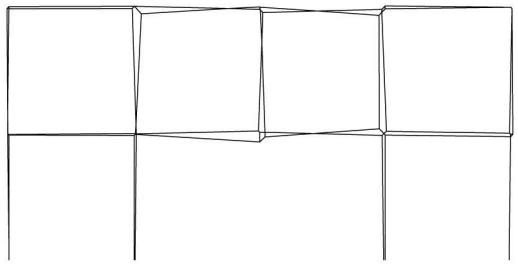
14 – 1: gd0.01 – Step 025: MD = 36.6 mm; Optimisation = 58%

14 – 2: gd0.01 – Step 158: MD = 25.1 mm; Optimisation = 72%

***Figure 14:** Topologies – Section B-B of* $[gd0.01]$



15 – 3



16 - 1

15 – 1: cm01 – Step 02: MD = 36.4 mm; Optimisation = 58%

15 - 2 : cm01 – Step 08: MD = 14.5 mm; Optimisation = 83%

15 – 3: cm01 – Step 16: MD = 12.2 mm; Optimisation = 86%

***Figure 15:** Topologies – Section B-B of* $[cm01]$

16 - 1: gd0.001 – Step 451: MD = 37.3 mm; Optimisation = 57%

***Figure 16:** Topologies – Section B-B of* $[gd0.001]$

5.3.3 PERFORMANCE ASSESSMENT

Compared to the structures of the gradient descent series $[gd0.01]$ and $[gd0.001]$, the overall geometry of the optimised cells appears more regular and symmetric than the geometry resulting from the gradient descent, which appears more adequate to the simple symmetrical system and the regular load case given.

Comparing the performance of $[cm01]$ with $[gd0.001]$ and $[gd0.01]$, it shall be pointed out that similar levels of optimisation are accomplished with comparably little effort in changes of the cell geometry. It appears that certain topologies which emerge from the combinatorial movements of adjacent Voronoi cell cores seem to generate statically rather efficient constellations: the symmetrical movements of adjacent cell cores let cells contort against each other whilst principal structural members of adjacent cells keep in touch with each other, thus maintaining structural integrity. These topologies enhance structural stability considerably from the first stages of the process onwards, even without significant geometric deformation of the structure. For example, the optimisation of 58% in the early stage of $[cm01]$ is reached by $[gd0.01]$ after 25 time steps, but with greater effort in terms of overall geometry deformation. $[gd0.001]$ reaches 57% optimisation with comparable little geometry change, but needs 451 iterations to get there (Figures 14 –16) .

In a nutshell, it appears that the strategy of exhaustively searching through combinations of Voronoi cell cores movements, despite various drawbacks[5] of the approach already mentioned, nevertheless seems to come up with interesting topologies, which accomplish structural improvement rates of 60% without major geometric changes in the early stages of the process.

In the next section, therefore, different types of topologies which emerge from the six different setups shall be analysed in more detail.

[5] It needs to be added to the list of drawbacks that the methodology presented here seems to become less efficient in the later stages of the optimisation process. In the late stages, the fitness is increased by only further 20% percents. The strongly restricted movement spectrum of each point fails to maintain and improve topologies as precisely as in the beginning of the process, due to cell geometries becoming more complex. Further research, however, could aim to develop an algorithm to specifically optimise the geometry of these topologies, but this is beyond the scope of this paper.

6 ANALYSIS OF THE VORONOI CELL TOPOLOGY AS A STATICAL SYSTEM

6.1 THE VARIETY OF TOPOLOGY FROM DIFFERENT SETUP TYPES

The six different setups $[cm01]$ to $[cm06]$ which are considered in this paper are developing considerable topological differences in the course of their evolution. As observed, these topological differences are already established in the first stages of the optimisation processes. The different topologies result in notable different force patterns – the way axial forces are distributed over the system -, and lead to considerably varying degrees of structural fitness among the results of each test case at similar stages – the worst performance after one iteration is $[cm03]$ with a maximum deformation (MD) of 67 mm (=24% optimisation), compared to a MD of 43.3 mm (= 51% optimisation) at $[cm04]$.

Figure 17 to Figure 22 shows the results of $[cm01]$ to $[cm06]$ after each one iteration – each group has been moved once.

After the first iteration, although Voronoi cell cores have only dislocated slightly out of the initial configuration, and thus overall *geometrical* changes of the Voronoi cells are minor, the *topologies* of the cell edges have already changed significantly. Typical topological features which emerge throughout all six setups are:

- cells are slightly contorted but principal structural members keep in touch to principal members of adjacent cells
- there are typical doubled edges, for example the horizontally spanning members of the centre cells which seem to engage in larger-scale configurations spanning throughout the structure
- edges move out of their initial straight horizontal and vertical arrangement and are bracing against each other

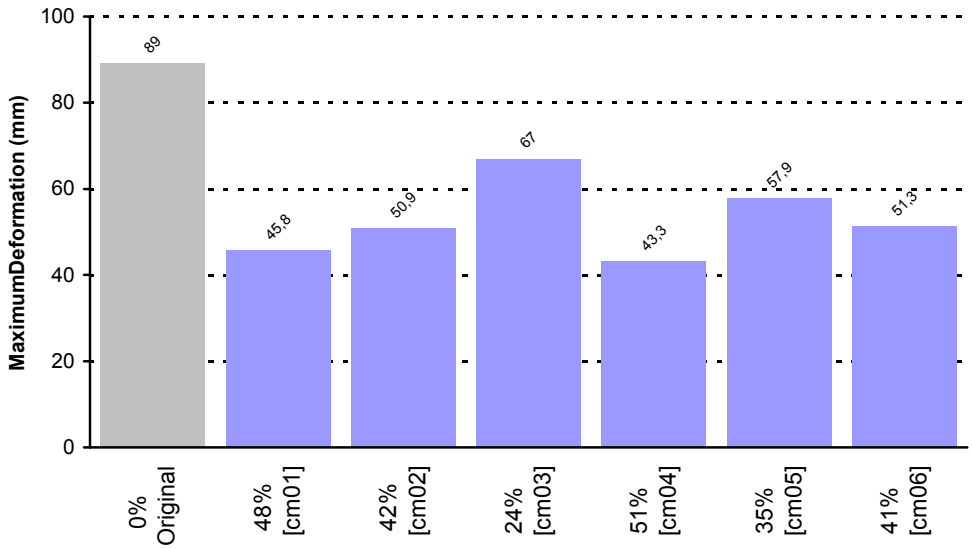
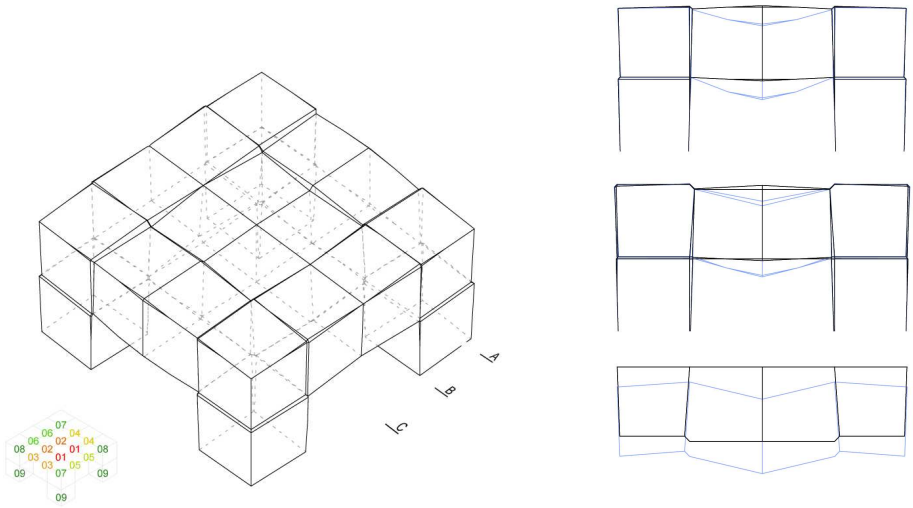


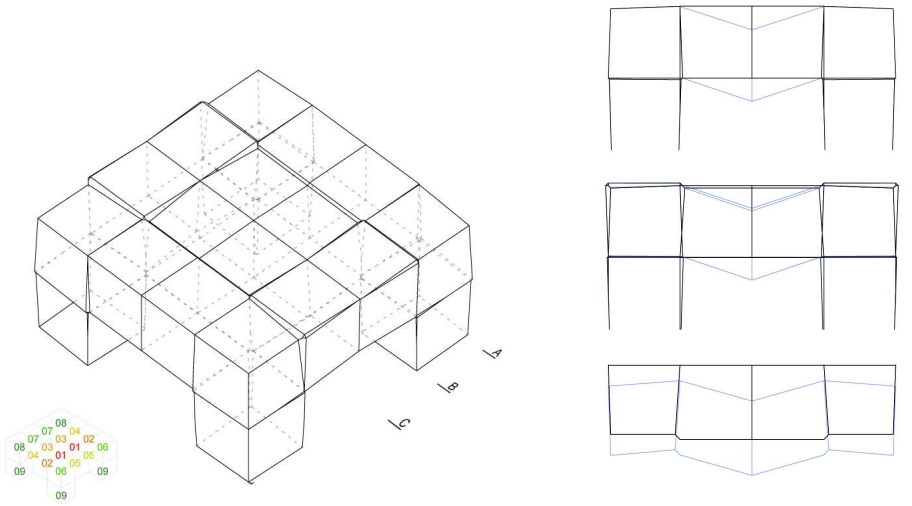
Chart 5: Maximum Deformation of $[cm01]$ **-** $[cm06]$ ***after 1 iteration***



MD=45.8 mm; Optimisation = 48%

Axonometry and sections A-A, B,B and C-C. Sections are overlaid with deformed sections.

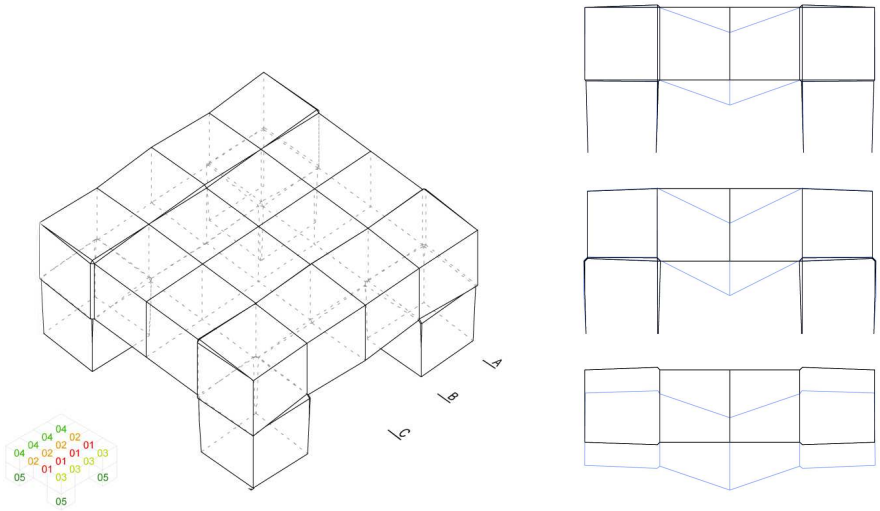
Figure 17: Optimised structure of $[cm01]$ ***after 1 iteration***



MD=50.9 mm; Optimisation = 42%

Axonometry and sections A-A, B,B and C-C. Sections are overlaid with deformed sections.

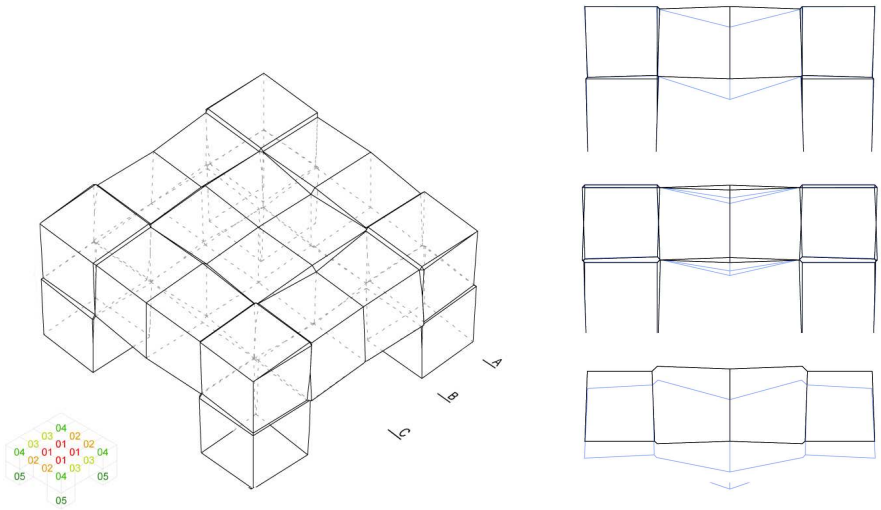
Figure 18: Optimised structure of $[cm02]$ ***after 1 iteration***



MD=67 mm; Optimisation = 24%

Axonometry and sections A-A, B,B and C-C. Sections are overlaid with deformed sections.

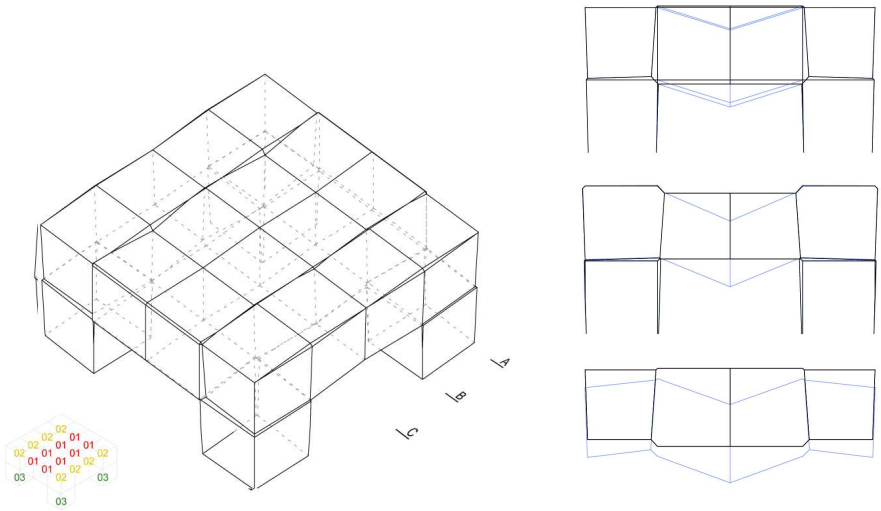
Figure 19: Optimised structure of $[cm03]$ ***after 1 iteration***



MD=43.3 mm; Optimisation = 51%

Axonometry and sections A-A, B,B and C-C. Sections are overlaid with deformed sections.

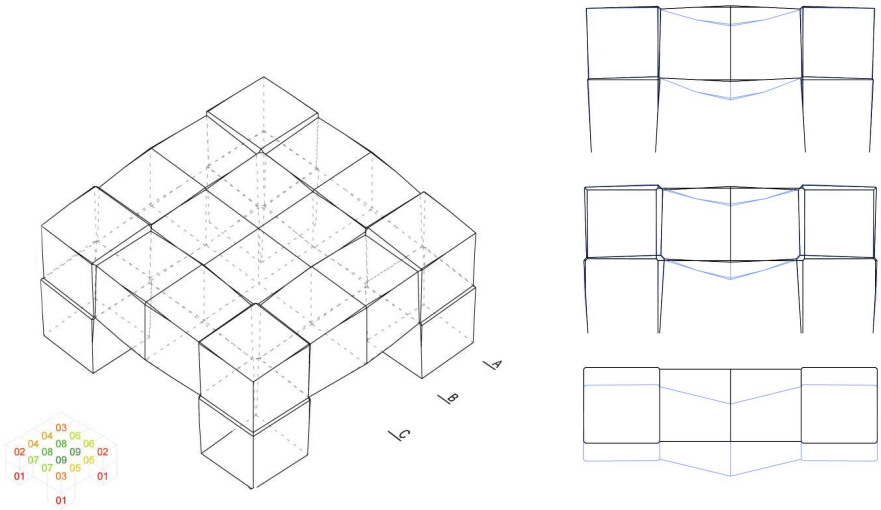
Figure 20: Optimised structure of $[cm04]$ ***after 1 iteration***



MD=57.9 mm; Optimisation = 35%

Axonometry and sections A-A, B,B and C-C. Sections are overlaid with deformed sections.

Figure 21: Optimised structure of [*cm*05] ***after 1 iteration***



MD=51.3 mm; Optimisation = 41%

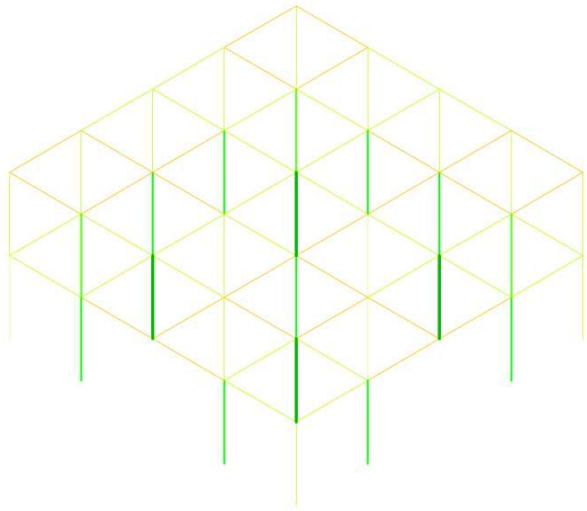
Axonometry and sections A-A, B,B and C-C. Sections are overlaid with deformed sections.

Figure 22: Optimised structure of [*cm*06] ***after 1 iteration***

6.2 ANALYSIS OF FORCE PATTERNS

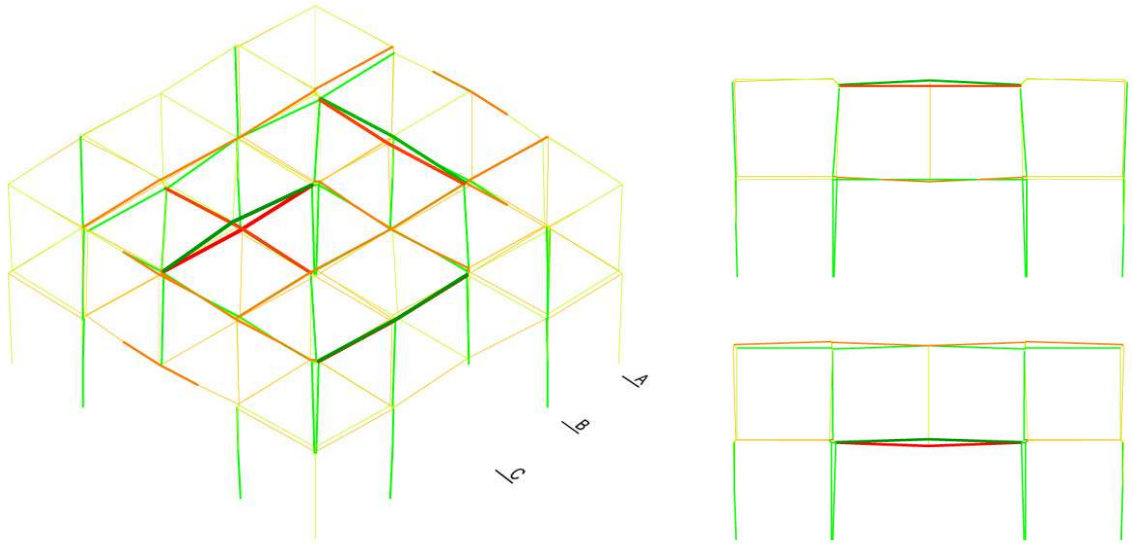
Each new topology has a distinct force pattern (Figures 23 – 32). Whilst the force pattern of the original structure is rather simple, featuring moderate pressure and tear forces (Figure 23), the optimised structures reveal force patterns which are much more complex (Figures 24 - 29). Coherent patterns of high stress are emerging which span over multiple cells. Furthermore, there are typical patterns such as constellations of tear and pressure forces in the doubled members in the highly stressed spanning centres of the structure, featuring each one pressure and one tear component. It appears that through these cooperating pressure and force-stressed members, the optimised structures are stabilising considerably. – In the later stages of optimisation, when structures tend to become more irregular and principal members loose connection to each other, these coinciding tear and pressure force constellations disappear again, and formerly coherent patterns of strong forces become more fragmented again (Figure 30).

In the early stages, however, these interacting pressure and tension forces seem to exhibit a distinct kind of structural quality which has a considerable effect on the statical fitness of the structure.



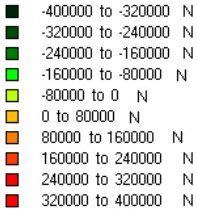
Axonometry

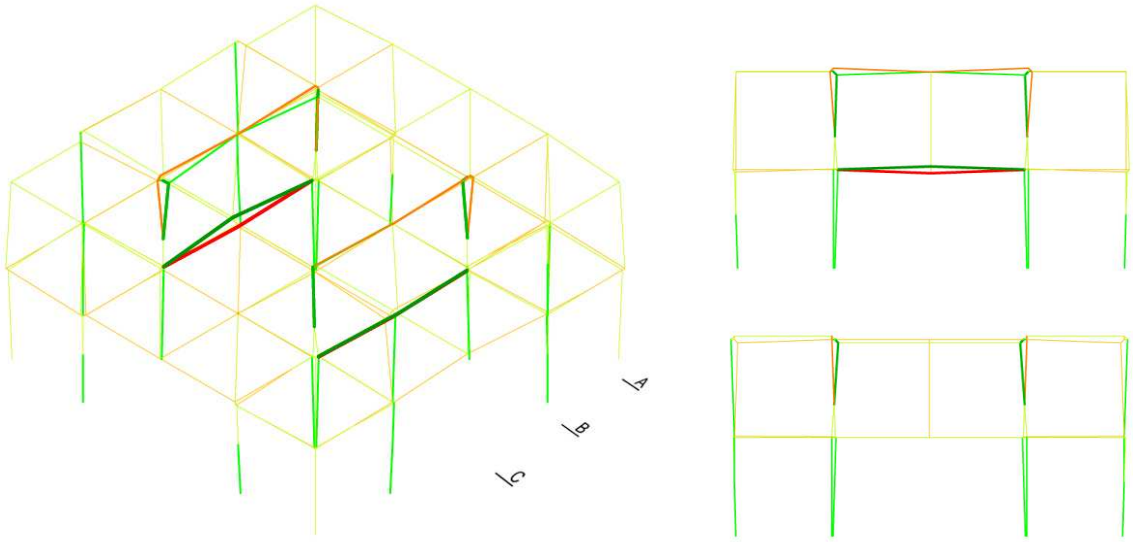
Figure 23: Original structure – Diagram of Axial Forces Fx



Axonometry and sections A-A and B-B

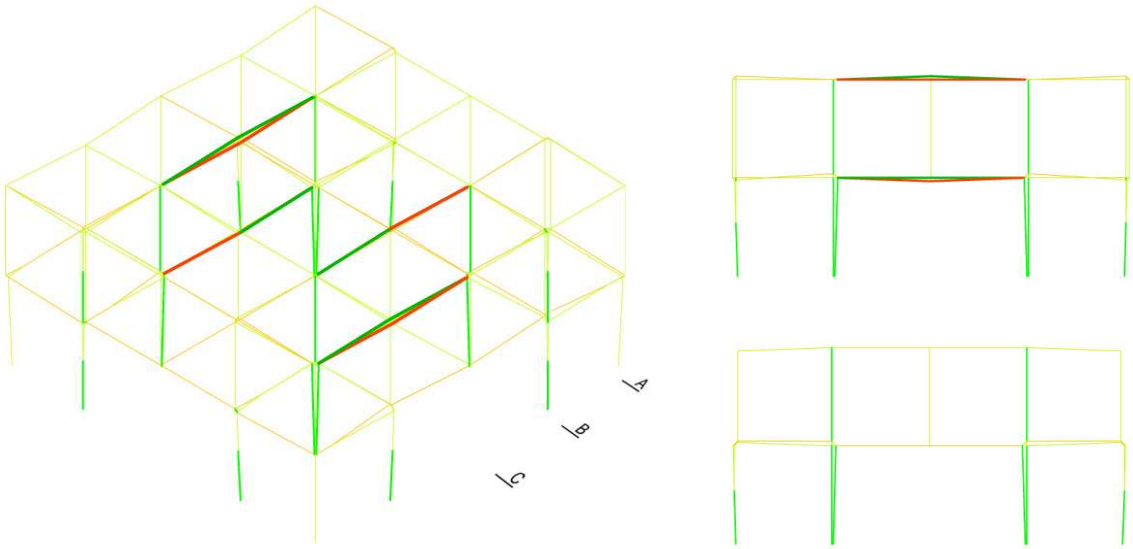
Figure 24: [*cm*01] ***1st iteration – Diagram of Axial Forces Fx***





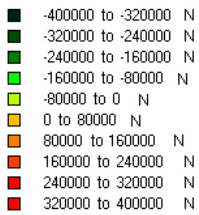
Axonometry and sections A-A and B-B

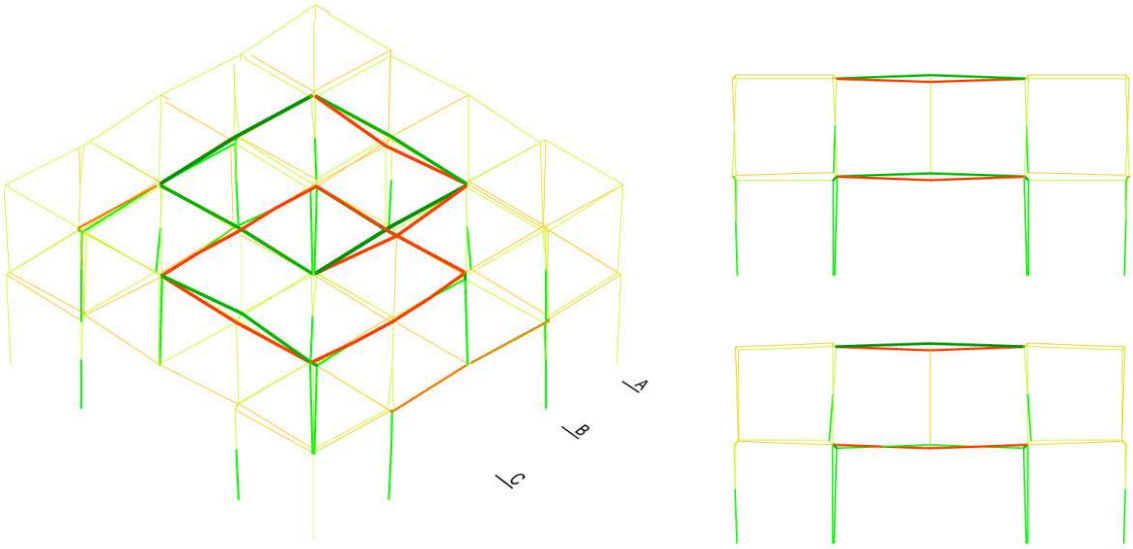
Figure 25: [*cm*02] ***1st iteration – Diagram of Axial Forces Fx***



Axonometry and sections A-A and B-B

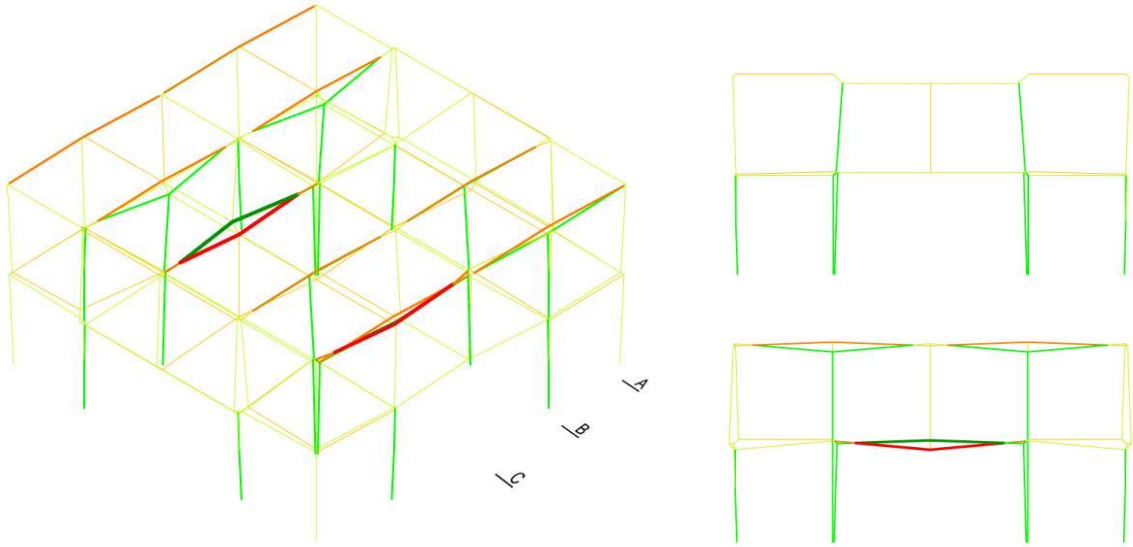
Figure 26: [*cm*03] ***1st iteration – Diagram of Axial Forces Fx***





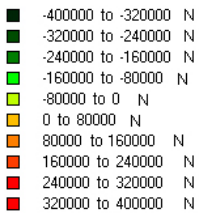
Axonometry and sections A-A and B-B

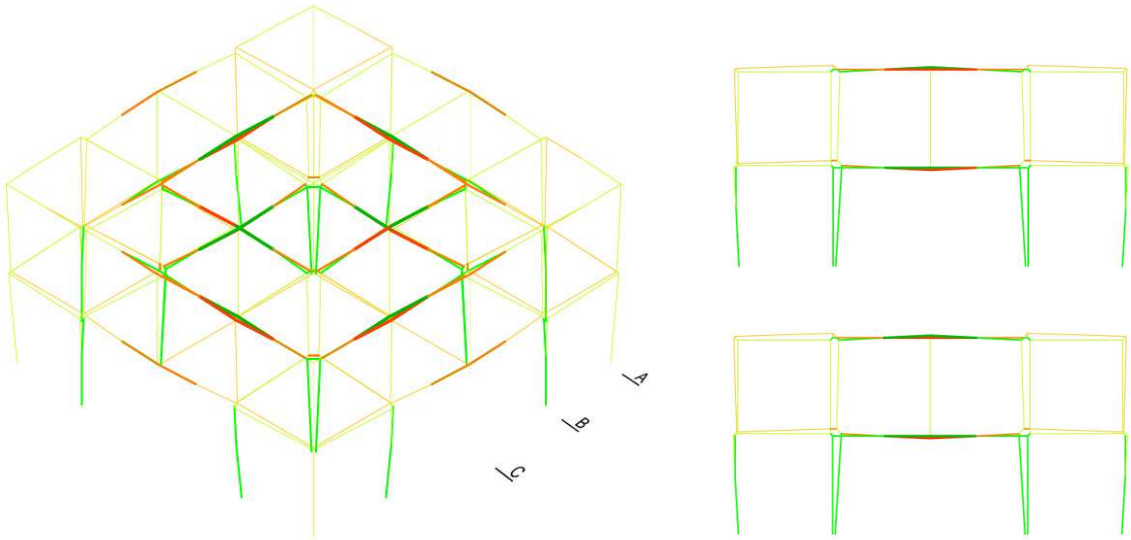
Figure 27: [*cm*04] ***1st iteration – Diagram of Axial Forces Fx***



Axonometry and sections A-A and B-B

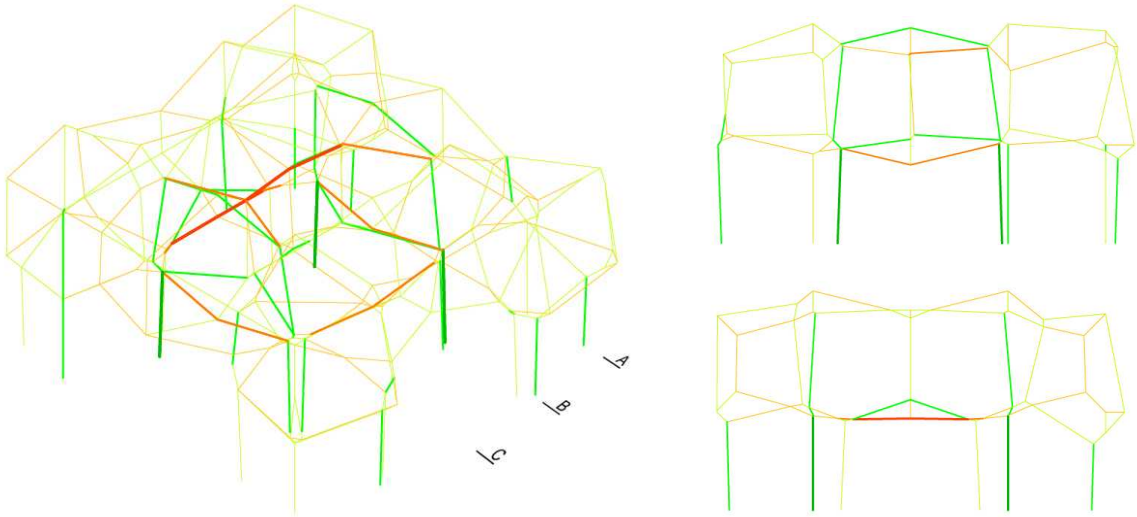
Figure 28: [*cm*05] ***1st iteration – Diagram of Axial Forces Fx***





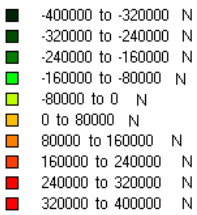
Axonometry and sections A-A and B-B

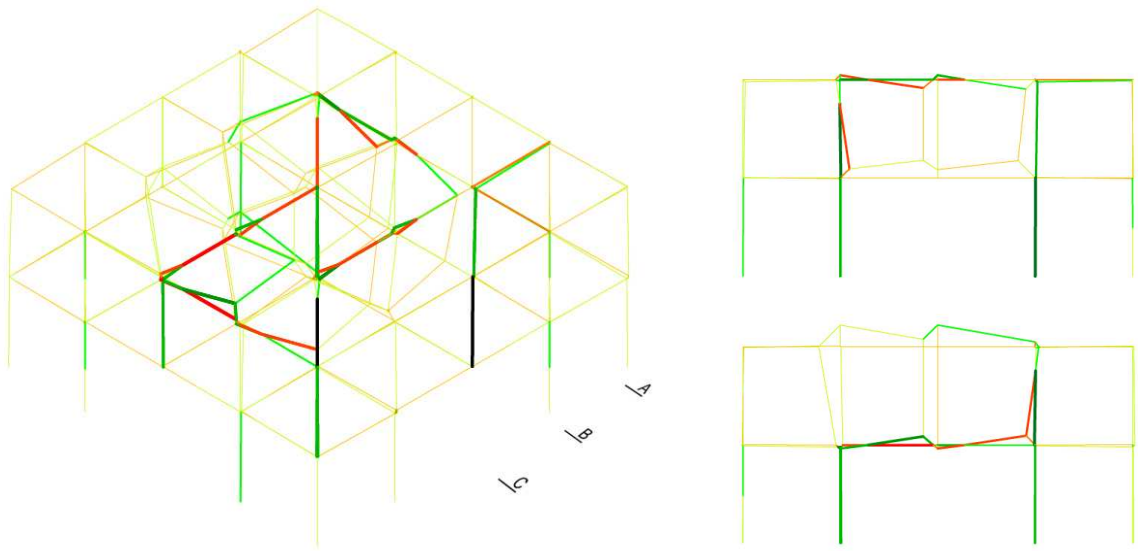
Figure 29: [*cm*06] ***1st iteration – Diagram of Axial Forces Fx***



Axonometry and sections A-A and B-B

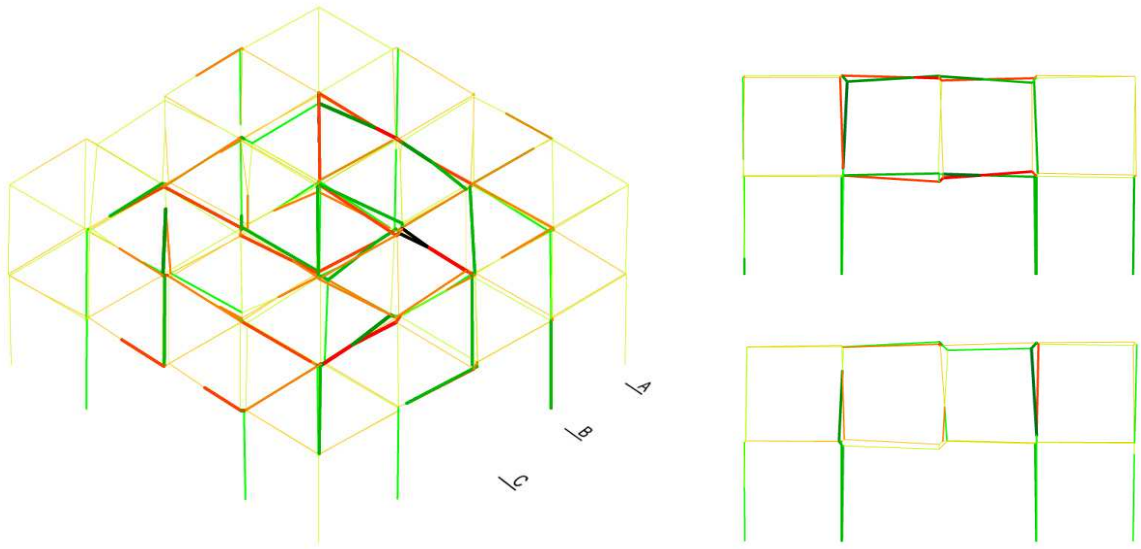
Figure 30: [*cm*01] ***16th iteration – Diagram of Axial Forces Fx***





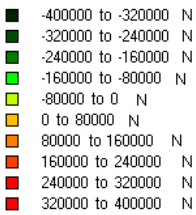
Axonometry and sections A-A and B-B

Figure 31: $[gd0.01]$ ***1st iteration – Diagram of Axial Forces Fx***



Axonometry and sections A-A and B-B

Figure 32 : $[gd0.001]$ ***491th iteration – Diagram of Axial Forces Fx***



Statical systems which exhibit strategic interactions of tear and pressure forces have been extensively analysed in their applicability to architectonic structural systems. Notably Buckminster Fuller observed that certain constellations of tear and pressure forces generate structural integrity which comes from the synergy between balanced tension and compression components. He coined the term ‘Tensegrity’ for this phenomenon (in: Meller, 1970). Buckminster Fuller explained the operating principle of such structures in the fact that tear and pressure are ‘not opposites, but fundamental physical laws that are complementally working together in a ‘win-win-relationship’’.

Although Tensegrity structures in their purest form obviously present an extreme type[6], ideas to exploit synergies of tear and pressure forces are nevertheless widely applied in architecture and structural engineering. The Swiss architect Valerio Olgiati, for example (Figure 33), used the synergy of tear and pressure members in the structure of the University of Lucerne, in order to make the statical structure ‘more efficient’ (Olgiati, 2006) and allowing the members being thinner in dimension. The design for the

[6] In classical Tensegrity structures, tension is continuous and compression discontinuous; pressure members are not interconnected to each other. This technique allows structures to be extremely lightweight. However, Tensegrity structures are sensitive to varying load cases and thus have been considered impractical for most real-world applications.

University of Lucerne was a winning contribution to a competition tendered in 2003. The statical structure, distorted and seemingly coincidental, is in fact precisely derived from static and functional preconditions. The slight contortion of the building is exploited to stabilise the structure in any three directions, with as few pillars as necessary. There are two types of pillars, the main load-bearing ones which are mainly stressed by pressure and which push up, accompanied by additional thin pillars which pull down at certain points where the horizontal beams cantilever and tend to bend upwards. This interplay of supporting and tearing elements 'makes the structure thinner and more efficient' (Olgiati, 2006), 'The building is a skeleton building, but on the other hand it is also an organic building, that is not modular anymore, even though it is based on the typology of a piloty system.'

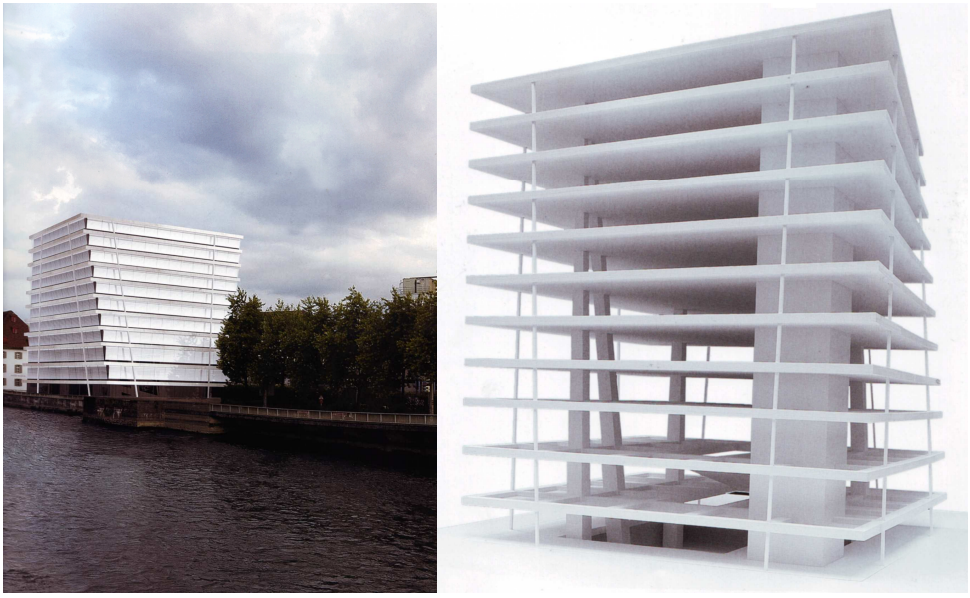


Figure 33: Valerio Olgiati: University of Lucerne. Competition entry, 2003

The Architects Meili & Peter and the structural engineers Juerg Conzett and Partners (Figure 34) developed a structural system for the lobby roof structure of the Swiss Re building in Rueschlikon, 1996-99 (Mostafa, 2006), which was specially designed to minimise deflection. The roof structure is ‘resolved’ into a system of tensile and pressure members, which **divert** forces more efficiently than straightforward beam members. The lobby is designed as a ‘garden hall’ with a glazed façade which should not be obstructed by columns. Thus the roof is cantilevering freely more than 13 metres from the rear wall of the lobby. However, the glazed façade is very sensitive to even slight deflections of the roof, and a standard timber beam construction would not have met these affordances. To minimise and reverse deflection and to release the façade from pressure forces, the structural engineers developed a technique which reversed and redirected the forces in a manner which transformed the system from a simple cantilevering beam structure into a more complex system joining together the various tension and pressure members as crucially interacting cooperative parts. The roof beams were post-stressed in a manner that they would tend to bend upwards, using the rear lobby wall as a pylon, with the cables wrapping over a ‘baffle saddle’ before disappearing back inside the beams above the hall. The beams take up the compression forces generated by the tension cables. Conversely, the permanent load of the beams and additional roof load are taken up by the tension cables. The façade itself is under tension instead of pressure, even in cases of maximal snow load. In the façade construction, thin rods under tension could so replace columns under compression.

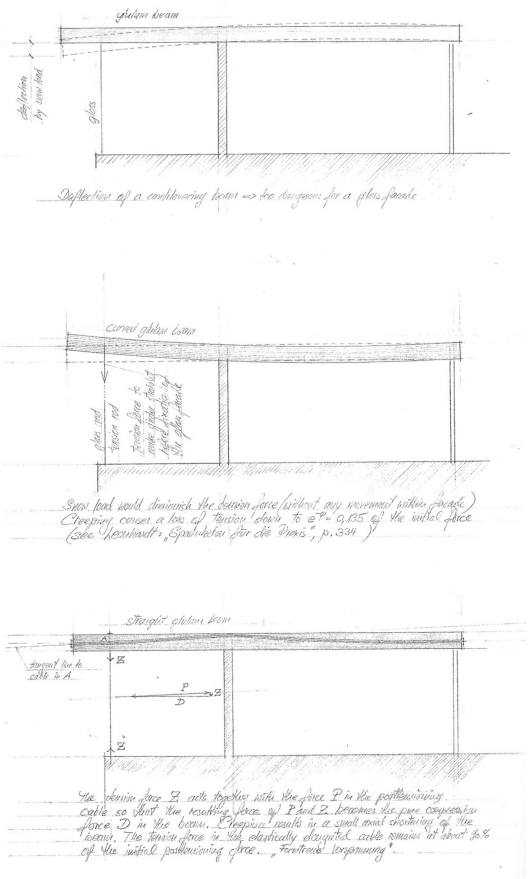


Figure 34: Meili + Peter, with J. Conzett: Swiss Re Lobby roof structure

Likewise, as shall be suggested, some of the structures presented in this section feature topologies of a structural integrity which allows synergetic tear and pressure forces to take place. Forces are directed more efficiently than in the original simple structure, and also more efficiently than in comparably stronger fragmented, but geometrically stronger deformed structures of the later stages of the optimisation process, or results of other optimisation techniques such as in $[gd0.001]$ and $[gd0.01]$ (Figures 30, 31, 32).

The effect of tear and pressure forces hereby can be read off the force patterns of the structural members. The variations in the degree of structural fitness of the results of $[cm01]$-$[cm06]$ suggest that coherent patterns of highly stressed members reaching throughout the structure are beneficial for minimising deformation. The best option, $[cm04]$, exhibits a constellation of high-stressed structural members which form a closed ring around the critical spots of maximal deformation, and efficiently transfer forces towards the vertical members. Other structures, in which force patterns are continuously spanning across the structure – as in $[cm01]$ - exhibit a good performance also, whereas, as in $[cm03]$ forces are non-coherent, inefficiently distributed and not tackling the actual problem zone; structural performance is less good and the maximum deformation is comparably higher – 67mm compared to the best result – 45.8mm in $[cm04]$. Structures generally perform well when they are not too fragmented, and crucially stressed members are well interconnected to the overall structure.

6.3 ANALYSIS SUMMARY

These observations indicate that structural performance of the optimised Voronoi cell structures is highly dependent on its capacity to act as a *system* of interconnected structural members. Structural performance does not add up linearly from the amount of material used and the mere sum of physical properties of material strengths, but emerges from the integrity of the structure as a whole.

It appears that a strength of the Voronoi diagram lies in the potential to produce interesting and unexpected structures which exhibit statically efficient system behaviour. Certain topologies which have emerged during the optimisation process apparently allow for ‘synergy’ effects of coordinated interactions of tear and pressure forces.

On the other hand, for the same reason that topology is crucial for performance, it appears that the structural fitness of the Voronoi structure can probably not be enhanced efficiently through a bottom-up ‘probabilistic’ approach which ‘guesses’ on the fitness landscape on the basis of out-of-context isolated cell core movements, which gives little control over the actual topology of the structure. As synergetic structural effects as observed in the structures presented in this section, however, apparently demand precisely connected structural layouts, the Voronoi diagram as a statical structure can probably only explored by deliberately iterating through combinations of Voronoi cell core movements and their emerging geometries.

7 FUTURE DEVELOPMENT

As has been argued above, the interesting potential of the Voronoi diagram for structural optimisation lies in its capacity to produce a multitude of different topologies and geometries, some of them leading to novel and unexpected structural solutions for statical optimisation problems. The Voronoi diagram hereby manipulates structural geometry and topology simultaneously - in contrast to strategies which either optimise topologies or geometries of predefined topologies, which, on the other hand, makes the emergent topology very difficult to control.

It has been indicated that the Voronoi diagram as a statical structure thus can probably only be explored by exhaustive search, which is computationally expensive and not feasible for more complex systems. This issue, however, would demand to be addressed in future developments.

This could be done either by increasing computing power, or by refining the technique to minimise computing cost as such, by combining exhaustive search with additional algorithms.

The computing power could be easily increased through a setup which distributes the necessary computing over a network, engaging several computers at a time: The exhaustive search could be easily done in parallel processing mode, as the state evaluations of the search space are independent processes.

In order to minimise computing demand one could combine the exhaustive search with other optimisation strategies in order to make the process more efficient: It has been observed that performative topologies can already be found in early steps of exhaustive search optimisation. These topologies

tend to persist through later stages. One could take advantage of this by letting the Voronoi diagram only evaluate the topology and then employ more suitable algorithms to optimise the actual geometry for this topology. At this stage, the Voronoi diagram could be dismissed. Instead, the new topology could be processed straightforwardly, for example as a connection graph of beams and nodes.

8 SUMMARY

This paper explores strategies to optimise a three-dimensional Voronoi cell structure as a statical system.

In order to act as a statical system, the edges of the Voronoi cells are considered as beam members with rigid connections, and have been assigned appropriate real-world material properties. The structure is assessed by statical analysis software in terms of the occurring maximal deformation – which is to be minimised during optimisation-, and seeks to optimise through amendments of the position of its Voronoi cell cores.

It has been pointed out that the relationship between the configuration of the Voronoi cell cores, the microstate of the system, and its resulting overall cell geometry, the system's macrostate, is inherently complex: each cell vertex is a combinatorial product of all adjacent Voronoi cell cores and thus depends on multiple parameters. Any change of the Voronoi cell core configuration triggers changes of the geometry of the cell structure as well as its topology instantaneous. Topological changes of the cell structure however can result in abrupt changes of the system's 'fitness' as a statical structure. In that sense, the degree of structural fitness in terms of the problem considered here is non-linearly related to the actual amount of dislocation of the Voronoi cell cores. The overall cell geometry *emerges* from the underlying Voronoi cell core configuration and is thus difficult to control precisely from the bottom – up – by amending the positions of the Voronoi cell cores.

This paper has explored two different techniques to optimise the Voronoi structure:

The first approach attempts to explore the gradient of the fitness landscape of the Voronoi system, given the systems states as the configurations of the Voronoi cell cores and the fitness of each state being the structural performance of the Voronoi cell geometry. The technique attempts to approximate the direction of the steepest ascent of the fitness landscape at each current state through evaluating individual test steps of individual Voronoi cell cores and merging them to a combined step, weighted according to the success of each test step.

It has be suggested that this technique is inherently flawed from the lack of control it provides over the topology of the cell geometry. As the specific topology of the Voronoi structure emerges from the respective constellation of the Voronoi cell core set, conclusions on a beneficial next state of the system may not be generated by adding up individual cell core movements. Specific structural features from individual moves might not happen in the combination of movement of neighbouring cell cores. In fact, optimised structures generated with this technique, although performing better than the original structure, look irregular and random, with considerable distortion of the geometry of the structure.

The second approach attempts to explore the emergent topology of the Voronoi structure more exhaustively. It enumerates solutions from combined movements of Voronoi cell cores and evaluates each in terms of its structural fitness. Exhaustive search, however, is computationally extremely expensive due to the abundant size of the search space, so it was only possible to search through movement combinations of subsets of cell cores which shrinks the search space to a feasible size. – Although the curtailment of the search space presents an obvious drawback of the method, it still appeared that the optimisation process was able to discover structures with a very specific structural potential. Certain topologies which emerged from combined movements of neighbouring cell cores – especially

in the early stages of the process – appear to generate ‘synergetic’ constellations of structural members, which, whilst cells are only minimally deformed, redirect forces more efficiently than the straightly connected structural members of the original cubic cell structure, or the more distorted structures from the earlier approach. It shall be suggested that this system behaviour of tear and pressure forces inherently emerges from the specific topology of the structure.

It has been suggested that the Voronoi structure can be considered as a complex system. The source of complexity hereby lies in the combinatorial explosion of the number of possible cell geometries, emerging from specific constellations of Voronoi cell cores. This type of complexity is difficult to control as it can neither be described by probabilistic means nor exhaustively searched due to its abundant possibilities of states.

This lack of control will limit the applicability of the Voronoi structure as a shape-generating tool for architecture and structural design problems. On the other hand, however, it shall be suggested that it is precisely this variety of novel geometries and topologies ready for exploration which bears a potential for unexpected solutions which would not have been detected otherwise.

REFERENCES

Sunny Y. Auyang, 1998: *Foundations of Complex-systems Theories*. Cambridge University Press, United Kingdom

T. Bonwetsch,S. Gmelin,B. Hillner,B. Mermans,J. Przerwa,A. Schlueter,R.Schmidt, 2006: *m-any*. www.m-any.org [Accessed August 2008]

M.Braach in collaboration with Kees Christiaanse Architects and Planners (KCAP), 2006: *Kaisersrot*. www.kaisersrot.com [Accessed August 2008]

Pablo Miranda Canzarra, 2001: *Self-design and Ontogenetic evolution*. In: Generative Art International Conference, Milan, Italy http://www.generativeart.com/on/cic/ga2001_PDF/miranda.pdf [Accessed August 2008]

Eva Friedrich: *Self-organising Room Layout using Kohonen Neural Network & 3D Voronoi Diagrams*, in: Fatah gen Schieck, A and Hanna, S (editors), 2007: *Embedded, Embodied, Adaptive: Architecture and Computation*. Emergent Architecture Press: London

Friedrich, E. and Hanna, S. and Derix, C., 2007: *Emergent Form from Structural Optimisation of the Voronoi Polyhedra Structure*. In: Generative Art 10th International Conference 2007, 12, 13, 14 December 2007, Milan, Italy.

R. Buckminster Fuller, 1956: *Octet truss patent*, United States Patent Office # 2,986,241 OCTET TRUSS).

http://ecosyn.us/Patents/patent2986241/octetruss_patent.html [Accessed August 2008]

Toyo Ito, 2006: *Blurring Architecture 1971-2005*. Rethinking the relationship between architecture and media. Publisher: Charta

John McHale, 1962: *R. Buckminster Fuller*. New York

James Meller (Editor), 1970: *The Buckminster Fuller Reader*, London

Mohsen Mostafavi (Editor), 2006: *Structure As Space: Engineering and Architecture in the Works of Juerg Conzett and His Partners*, p. 093-094. AA Publications, London, United Kingdom

Oasys GSA. www.oasys-software.com

Valerio Olgiati, 2006: *Inventioneering Architecture*. Lecture accessible at: www.architecture-radio.org/inventioneering [Accessed August 2008]

Frei Otto et al, 2008: *Ausgewaehlte Arbeiten von Frei Otto und seinen Teams.* http://www.freiotto.com/ [Accessed August 2008]

Processing: www.processing.org

Warren Weaver, 1948. *Science and complexity*. American Scientist 36: 536.

Eric Weisstein: *Exhaustive search*, http://mathworld.wolfram.com/ExhaustiveSearch.html [Accessed August 2008]

Wikipedia the free encyclopedia: *Complex Systems*. http://en.wikipedia.org/wiki/Complex_system [Accessed August 2008]

Wikipedia the free encyclopedia: *Disorganised Complexity vs. Organised Complexity*.

http://en.wikipedia.org/wiki/Complexity#disorganised_complexity_vs._organised_complexity [Accessed August 2008]

Wikipedia the free encyclopedia: *Gradient descent*. http://en.wikipedia.org/wiki/Gradient_descent [Accessed August 2008]

Wikipedia the free encyclopedia: *Exhaustive search* http://en.wikipedia.org/wiki/Exhaustive_search [Accessed August 2008]

Wikipedia the free encyclopedia: *Tensegrity.* http://en.wikipedia.org/wiki/Tensegrity [Accessed August 2008]

Stephen Wolfram, 2002*: A New Kind of Science*, Wolfram Media, Inc. http://www.wolframscience.com/nksonline [Accessed August 2008]

APPENDIX

1 MAIN COMPONENTS AND VARIABLES

```
1       Vector Cores;
```

The set of Voronoi points is represented by a vector of the class `Core`. `Core` holds the position information of each point as well as additional information such as the set of neighbours etc.

```
2       DelaunayTetrahedrisation dt;
```

The class for the tetrahedrisation

```
3       Vector bars;
        Vector nodes;
```

Bars and nodes represent the structural beams and interconnections. Geometrically, they are in principle matching the Voronoi cell edges, however, they are clipped on the bottom plane.

2 GENERATING THE STATICAL STRUCTURE

```
void createBars() {

  bars=new Vector();

  nodes=new Vector();

  barcounter=0;

  nodecounter=0;
```

```
//loop through all tetrahedrons
for (Iterator it = theTetrahedrisation.iterator(); it.hasNext();) {
  Simplex triangle = (Simplex) it.next();
  //loop through all facets of each tetrahedron
  for (Iterator It2 = triangle.facets().iterator(); It2.hasNext();) {
    Set facet = (Set) It2.next();
    Core[] endpoint = (Core[]) facet.toArray(new Core[3]);
    //contains the facet a core which is part of the statical structure?
    int sum=0;
    for (int i = 0; i < endpoint.length; i++) {
      if (endpoint[i].typ==2) {
        sum++;
      }
    }
    if (sum>0) {            //if so
      //find the neighbour of this facet
      for (Iterator otherIt = theTetrahedrisation.neighbors(triangle).iterator();
                   otherIt.hasNext();) {
        Simplex other = (Simplex) otherIt.next();
          if (other.containsCore(endpoint[0].id) &&
                   other.containsCore(endpoint[1].id) &&
                   other.containsCore(endpoint[2].id)) {
```

```
//a cell vertex joins the circumcentres of two adjacent tetrahedrons

//thus: find circumcentres
Pnt p = Pnt.circumcenter(triangle);

Pnt q = Pnt.circumcenter(other);

//create bar

 bar ba;

Vector sharedfacetCores = new Vector();

for (int i = 0; i < endpoint.length; i++) {

  sharedfacetCores.addElement(endpoint[i]);

}



ba = new bar(p,q,sharedfacetCores,barcounter+1);

if (!containsBar(bars,ba)) {

   //remove the bar if it is below the ground plane

  ba=removeIfOffStage(ba);

  if (ba!=null) {

    //clip bar on bottom plane

    ba.clipBar();

    //create nodes which joins bar to other bars

    ba.createNodes();

    if (ba.nodeoneend.id!=ba.nodeotherend.id) {
```

```
                    bars.addElement(ba);
                    barcounter++;
                  }
                }
              }
            }
          }
        }
      }
    }
  }
}
```

3 EXPORTING BARS AND NODES TO OASYS GSA

```
void exportBars(String path) {
  Vector string_nodes = new Vector();
  Vector string_spc = new Vector();
  Vector string_bars = new Vector();
  String s;
  //for node export
```

```
for(int i = 0; i<nodes.size();i++) {
  node n = (node)nodes.elementAt(i);
  s = "NODE\t"+n.id+"\t"+n.pos.coo[0]/10
            +"\t"+n.pos.coo[1]/10+"\t"
            +((stagez/2)-n.pos.coo[2])/10
            +"\t0";
  string_nodes.addElement(s);
  //for restrictions export  - fixed nodes on bottom plane
  if (n.isRestricted) {
    s="SPC\t"+n.id+"\t0\t1\t1\t1\t1\t1\t1";
    string_spc.addElement(s);
  }
}
//for beam export
for (int i = 0; i < bars.size(); i++) {
  bar b = (bar)bars.elementAt(i);
  s="EL_BEAM\t"+(b.id)+"\t1\t1\t"
            +b.nodeoneend.id+"\t"
            +b.nodeotherend.id
            +"\t0\t0.000000";
  string_bars.addElement(s);
```

```
  }

  //create textfile for text export
  String list[] = new String[string_nodes.size()

              +string_spc.size()

              +string_bars.size()+3];

  [ . . . ]



  //add material property for beam

  list[counter]="SEC_BEAM\t1\tSection 1\tSTEEL
        STD%CHS%300.%20.\tNA\t0.000000";

  //add loading (self-weight)

  list[counter+1]="LOAD_GRAVITY\tall\t1\t0.000000\t0.000000\t-1.00000";

  //finalise

  list[counter]="END";

  //save

  saveStrings(path, list);

  //analyze: run external program to run GSA analysis

  open(GSAlinker);

}
```

4 IMPORT ANALYSIS RESULTS

```
boolean importResults(String path) {

  gsaimportstring = loadStrings(path);

  String disp = "DISP";

  String moment = "REACT_MOMENT";

  for (int i=0; i < gsaimportstring.length; i++) {

    String l = gsaimportstring[i];

    //read displacement of nodes

    if (l.length()>=disp.length()&& l.substring(0,4).equals(disp)) {

      String[] ss = l.split("\t");

      int nodenum=Integer.valueOf(ss[1]).intValue();

      double dx = Double.valueOf(ss[3]).doubleValue();

      double dy = Double.valueOf(ss[4]).doubleValue();

      double dz = Double.valueOf(ss[5]).doubleValue()*(-1);

      Pnt cd = new Pnt(dx,dy,dz);

      //store maximum occurring displacement

      if (cd.length() > globaldisp) globaldisp=cd.length();

      //store node's displacement

      node n = getNode(nodes,nodenum);

      n.displaced = new Pnt(cd);

    }
```

```
  }

  return true;

}
```

5 TECHNIQUE 1: GUESSING ON THE GRADIENT OF THE FITNESS LANDSCAPE

```
//a Vector to hold the fitness increase of all test steps
Vector gradients;

//after each test step, save the degree of optimisation in a vector

//this will produce a vector of the size of test steps done

//in this context: (number of cell cores) * (dimensions) = 30 * 3

Double d;

//get previous displacement from the Vector MaxDisps()
double last = ((Double)MaxDisps.elementAt(MaxDisps.size()-1)).doubleValue();

//get degree of optimisation achieved in the test step just done
d=new Double(last-currentDisp);

//save it
gradients.addElement(d);

//after all test steps are done

//move all cell cores according to the fitness increase achieved by their test
steps

for (int i = 0; i < Cores.size(); i++) {

   Core b = (Core)Cores.elementAt(i);
```

```
   If (!b.structural) return;

   double x = ((Double)gradients.elementAt(i*3)).doubleValue();

   double y = ((Double)gradients.elementAt(i*3+1)).doubleValue();

   double z = ((Double)gradients.elementAt(i*3+2)).doubleValue();

   Pnt p = new Pnt(x,y,z);

   p.scale(0.001);

   //move the cell core to the new position

   b.pos.add(p);

}
```

6 GSALINKER: CALLING THE GSA COM INTERFACE

```
void RunGSAExportFunction(Cstring* csPath1)

{

        COleDispatchDriver cGsaDispDriver;

        cGsaDispDriver.m_bAutoRelease = true;

        BYTE      pArgType1[] = "";

        BYTE      pArgType2[] = VTS_BSTR;

        //init com

        bool init = AfxOleInit();

        if (!init)

        {

                AfxMessageBox("Error when Initialising COM.");

        }

        AfxEnableControlContainer();
```

```
    CoInitializeEx(NULL,COINIT_MULTITHREADED);
    // Create an instance of the GSA class "ComAuto"
    COleException e;
    if(!cGsaDispDriver.CreateDispatch("Gsa.ComAuto",&e))
    {
            AfxMessageBox("GSA not found or not registered");
            return;
    }
   //analyse
    bool bStat(true);
    // Function Open
    if(bStat)
bStat = RunOneFunction(&cGsaDispDriver,
                                    "Open",
                                    pArgType2,
                                    csPath1);
    // Function Delete
    if(bStat)
bool bStat2 = RunOneFunction(&cGsaDispDriver,
                                    "Delete",
                                    pArgType2,
                                    csDelete1);
```

```
        // Function Analyse
        if(bStat)
        bStat = RunOneFunction(&cGsaDispDriver,
                                    "Analyse",
                                    pArgType1,
                                    "");
        // Function SaveAs
        if(bStat)
    bStat = RunOneFunction(&cGsaDispDriver,
                                    "SaveAs",
                                    pArgType2,
                                    csPath1);
        // Function Close
        if(bStat)
    bStat = RunOneFunction(&cGsaDispDriver,
                                    "Close",
                                    pArgType1,
                                    "");
}
```

```
bool RunOneFunction(
                COleDispatchDriver*      pDispDriver,
                CString                  csFuncName,
                BYTE*                        pArgType,
                CString                  csArgument)
{
                        DISPID      dispid;
                        OLECHAR*      pcsFunc;
                        CString      csMsg;
                        int             iReturn(0);
                        pcsFunc = csFuncName.AllocSysString();
//Find the function ID
if(pDispDriver->m_lpDispatch->GetIDsOfNames(
      IID_NULL,
      &pcsFunc,
      1,
      NULL,
      &dispid) != S_OK)
{
      csMsg.Format(
            "Function (%s) cannot be found",
            csFuncName);
      AfxMessageBox(csMsg);
      return false;
}
```

```
//Run the function
if(csArgument.IsEmpty())
	pDispDriver->InvokeHelper(
		dispid,
		DISPATCH_METHOD,
		VT_I2,
		&iReturn,
		pArgType);
else
	pDispDriver->InvokeHelper(
		dispid,
		DISPATCH_METHOD,
		VT_I2,
		&iReturn,
		pArgType,
		csArgument);


if(iReturn != 0)
	return false;
else
	return true;
}
```