Advanced Engineering Informatics 46 (2020) 101162

Contents lists available at ScienceDirect

Advanced Engineering Informatics

journal homepage: www.elsevier.com/locate/aei

Spatial zoning for better structural topology design and performance

Dennis P.H. Claessens, Sjonnie Boonstra*, Hèrm Hofmeyer

Eindhoven University of Technology, the Netherlands

ARTICLE INFO

Keywords:
Building spatial design
Structural design
Stabilization
Zoning

ABSTRACT

A new three-dimensional spatial zoning procedure that has been tailored to structural design is presented in this work. Zoning is the grouping of spaces or sub-parts of spaces such that the resulting spatial layout is logical from a disciplinary perspective. Two automated structural design methods, so-called grammars, have been defined to generate structural designs for zoned building spatial designs. The first grammar, here termed stable design grammar, places structural slabs at the boundaries of a zone, and columns at places where a zone intersects with a space. The second grammar, here termed unstable grammar, generates a structural frame that is unstable by definition. Via a new stabilization method, which is also presented in this work, the unstable structural frame is stabilized by considering the boundaries of a zone for the placement of stabilizing measures. In a case study, the zoning procedure is applied in conjunction with each of the design grammars to three distinct building spatial designs. The case study shows—for both design grammars—that structural designs use less material and are stiffer when they are generated based on a zoned building spatial design. Moreover, the zoning procedure has been tailored such that it yields a subset of all possible zoned designs for which, in general, the generated structural designs perform well.

1. Introduction

Structural design of buildings has remained a human task in a building industry that is automated more and more. This is mainly due to the fact that engineers are able to apply a top-down approach in their design process, e.g. they can discover shapes and patterns in a Building Spatial Design (BSD) that are relevant for the structural design. Subsequently, they know how to use these shapes and patterns for their structural design. Automated methods that are developed to design a structure usually use a bottom-up approach, in which only one part of a BSD or a small collection of parts in a BSD together are considered at a time to generate a structure. In this work, a new three dimensional zoning procedure for BSDs is introduced in combination with two new automated structural design methods for zoned building spatial designs. Here, zoning is the grouping of spaces or sub-parts of spaces such that the resulting spatial layout is logical from a disciplinary perspective. The automated structural design methods are termed design grammars, and—in this work—are sets of rules that operate on part(s) of a BSD in order to generate a structural model. The zoning procedure and the grammars together form a method that treats the structural design of a BSD in a more top-down manner. A case study—also presented in this work—shows that applying a zoning procedure on a BSD before applying a structural design grammar to generate the structural design can improve the performance of the resulting structural designs.

This paper is structured as follows. First, the related work, background, and the motivation for the presented work are discussed in Section 2. Thereafter, in Section 3, a toolbox that was used and extended for the work in this paper is introduced and the parts that are relevant for this paper are explained in more detail. Following that, in Section 4, the new zoning procedure is presented, and in Section 5, the new structural grammars are outlined. Next, in Section 6, the case study is presented together with its results. The results are then discussed in Section 7, and finally, in Section 8, the conclusions and the outlook are given.

2. Related work and background

This section first discusses the related work on automated structural design generation and spatial zoning. Thereafter, the background and the motivation for the presented work are explained.

2.1. Automated structural design

Structural Design (SD) of buildings is strongly related to Building Spatial Design (BSD): A BSD can only exist or be experienced if it is realized by a structure. While at the same time, the structure occupies

* Corresponding author.
E-mail addresses: d.p.h.claessens@alumnus.tue.nl (D.P.H. Claessens), s.boonstra@tue.nl (S. Boonstra), h.hofmeyer@tue.nl (H. Hofmeyer).

https://doi.org/10.1016/j.aei.2020.101162
Received 30 November 2019; Received in revised form 21 July 2020; Accepted 20 August 2020
1474-0346/

space and affects the aesthetics of a BSD [21]. However, a discrepancy between the two exists with respect to the design stage during which the two are defined. The spatial design is conceived in a conceptual design stage, and often only after the BSD is fixed, its structural design is developed. Due to their strong inter-dependency, much of the structural design is already determined during the conceptual design. Wang et al. [35] and Brown and Mueller [5] see a similar trend and state that decisions in the early design stages of a building have the most effect on discipline specific designs of that building. Thus, the efforts spent at an early stage of design can determine to a large extent the performance of the final design. However, only a limited amount of discipline specific tools is available during these stages [25], whilst these tools are important in the exploration of design alternatives [28]. Another factor that limits the involvement of structural design in the conceptual design stage is the time that would need to be invested to consider every conceptual design alternative [13,12]. Therefore, automated structural design is promising, because it can provide designers with a tool that they can use to quickly assess the structural design potential of a conceptual building design.

In practice, the structural design of a building usually starts with at least a BSD as a starting point. A structural design engineer then determines the overall structural layout based on the spatial layout. Thereafter, the mechanical scheme of the structure system is conceived. This draft is then progressively further engineered while also the design of the building progresses, during which the materials, dimensions and construction of the building are determined (i.e. stress engineering). Because a structural building design is (usually) based- and dependent on a BSD, a BSD is required as input for an automated design process. In automated structural design, decompositions of the geometry of a BSD are made such that structural components can be placed at locations of the found geometric parts. Rafiq and MacLeod [26] presented a method that finds all the corner points of the BSD and accordingly uses these points to define the beams and slabs of a structural model. Geyer [14] defined design rules that each operate on a different piece of information in a BIM model, in conjunction with each other these rules generate a structural model. Boonstra et al. [3] use design rules to decide what type of structure (e.g. a beam, wind bracing, slab, or nothing) is assigned to each individual geometry part (such as a line segment or rectangle) in a BSD. Steiner et al. [32] create grid lines in a floor plan layout, which are then used as possible locations for structural components. Moreover, they have made their method interactive, such that a designer can change a resulting structural design. Smulders and Hofmeyer [31] realized that automated structural design methods did not explicitly take into account mechanical stability. Therefore, they developed an automated process that can stabilize structural design models by adding new components between existing points in these models.

When studying the methods for automated structural design, two aspects are observed. Firstly, in automated structural design an assumption seems to be made that structural components can only be placed at the boundaries between spaces, and sometimes this assumption is even made explicitly [11]. However, in many practical designs it is accepted for structure to interfere with space in order to improve the structural performance, e.g. beams in an atrium or a column in a space. Secondly, most methods position structural components based on the elementary parts of a building spatial design (e.g. a space or a surface), i.e. in a bottom-up approach. However, it is also possible to use a top-down approach, where structural components are positioned based on a decomposition of the spatial design. This decomposition can be found via two methods: either (a) it is built up from the elementary parts, or (b) it is derived from the complete building volume. Both are viable, and method (b) can be illustrated by a high-rise building, where the structural design is based on a decomposition of the building volume, and spaces are only assigned after the structural system is determined. For the automated structural design proposed here, the advantage of using method (a) is that the elementary (smallest) parts are used, and so a single generic method can be used to find all possible decompositions, which can accordingly be tested for structural performance. Method (b) may need another (tailored) method to make the decomposition for each specific building volume, and then it is still not guaranteed that all possible (structurally suitable) decompositions are found. Therefore, in this work, and for the application in automated structural design, method (a) is used. Finally, automated design methods for other disciplines that use a more or less top-down approach, such as presented by Dogan et al. [9], are generally tailored to that specific discipline. This means that the design method then only finds zoned designs with characteristics that are relevant to that discipline, and it is thus not well suited for other disciplines, e.g. structural design.

2.2. Zoning

In a BSD, the spatial layout is not necessarily logical for each design discipline [20]. Björk et al. [1] state that grouping spaces or sub-parts of spaces into zones can lead to spatial layouts that are more logical from a discipline's point of view. The literature provides examples of applications of zoned designs to HVAC design [6,9], day-lighting design [27], structural design [22], construction technology [29], a design tool for energy and daylight performance [8], and the generation of multiple spatial layouts from existing layouts to support collaboration between different design disciplines [33]. Moreover, gaps and requirements for the automatic generation of space layouts aimed at optimized energy performance have been presented by Du et al. [10]. For these examples, zones are sometimes obvious design elements, e.g. ventilation (in HVAC) design applies to a group of spaces that are openly connected to each other rather than to each space individually. However, for structural design, the use of zones may be less apparent, but nevertheless it is performed (subconsciously) by an engineer, e.g. when a column is placed in the center of a room. Procedures for spatial zoning aimed at structural design have been described using an automated two-dimensional approach in Ref. [19] (Fig. 1). Mora et al. [24] consider the possibility that users can assign structural zones within a BSD, and accordingly they use this information to automatically generate a structural design for each zone individually. Hofmeyer and Bakker [16] presented a fully automated three dimensional zoning algorithm which has been applied in a research engine that simulates a spatial-structural design process in Refs. [7,17]. They demonstrated that spatial zoning has a larger influence on the displacements of structural solutions than the amount of structural mass. Besides this, little is known about the influence of spatial zoning on the performance of resulting structural designs.

Zoning can be seen as a combinatorial problem, with possibly a large number of solutions. However, for efficient exploration, it is important that only the zoned designs that are relevant for structural design are considered [34]. Meyer and Fenves [23] and Parent et al. [25] learned from the zoning methodology that structural engineers apply. They conclude that structural engineers search for continuous planes—both vertically and horizontally—in a BSD in which the members of a structural system can be placed. When using space

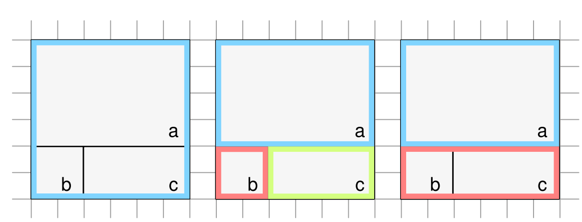

Fig. 1. Two dimensional spatial zoning of a floor plan with three spaces (indicated with letters); as presented in Ref. [19]. Three zoned designs are possible when a zone (red, blue, or green) is defined as a rectangle consisting of one or more whole spaces. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

boundaries to construct these planes (which is intuitive), these planes may be discontinuous, e.g. for the case where two spaces are adjoined at the mid-span of a space beneath. For this problem, zoning can help by using zone surfaces to construct the continuous planes, because: (a) zones can be allowed to intersect spaces, and (b), discontinuities may be bypassed by combining several spaces into a zone.

2.3. Background and motivation

The presented research in this paper is performed within a larger research framework that focuses on multi-disciplinary building spatial design optimization. Within this framework a toolbox has been developed [3,4], which contains tools that can be used to develop and research optimization methods, which are: (1) building spatial design representations, (2) automated design methods for structural design and building physics design, (3) analysis models for structural design and building physics, (4) data analysis tools such as clustering and non dominated sort, and (5) a visualization package for the different representations and models in the toolbox. Finding the best potential for any discipline for a given BSD is important for assessing a BSD's quality with respect to that discipline. This also applies to structural design, and this work therefore presents a zoning method that has been implemented in the toolbox, aiming to improve the designs resulting from the toolbox' automated structural design procedure. This automated structural design procedure can generate structural designs based on geometric and spatial relations, and can be configured to apply a range of different structural components (i.e. slabs, beams, and trusses) with any desired material properties. However, up until this work, the automated structural design procedure operated solely on the spatial boundaries. The proposed zoning method entails a three dimensional zoning method, that is conveniently incorporated with other tools that are already present in the toolbox. Note that, besides an application in multi-disciplinary building spatial design optimization, the zoning method can also provide design support to structural engineers and architects by helping them with design alternatives. Two new automated structural design methods that are able to handle zoned building spatial designs have been implemented in the toolbox: Method (a) places structural slabs on the boundaries of zones, but beams/columns where a zone boundary intersects a space, and as such a resulting structural design is guaranteed to be stable. Method (b) places structural trusses at the boundaries of spaces, resulting in an initially unstable structural design, which is then stabilized by considering the boundaries of zones for the placement of stabilizing components. To be able to resolve the structural instabilities in method (b), a new structural stabilization method has been implemented in the toolbox, which is similar to the method published by Smulders and Hofmeyer [31]. Both methods (a) and (b) are used in this work to investigate the influence of the different settings for the zoning procedure on the performance of the structural design. Moreover, method (b) is used to study the effects of zoning on the structural stability of a building. Finally, this paper proposes a zoning method that can generate a set of alternatives which can be reduced based on spatial features to a subset that is relevant to only structural design. In literature, these characteristics of zoning have only been encountered mutually exclusive, i.e. a method generates only one zoned design that is specific to a discipline, or a set of zoned designs that is not specific to a discipline. A set of zoned designs can be helpful, because it offers alternatives from which a designer or an optimization method can choose.

3. BSO toolbox

The work in this paper is performed using a toolbox for Building Spatial design Optimization (BSO), which is termed the BSO toolbox [3]. The BSO toolbox is an open-source C++ library that contains various tools for the evaluation of building spatial designs and provides the means to develop building spatial design optimization methods and the source code is available online [4]. Those parts of the BSO toolbox that are relevant for the methods described in this paper are explained in this section, beginning with the definition of a Building Spatial Design (BSD). Thereafter, a decomposition of a BSD into a so-called building conformal model is explained. And finally, the generation and evaluation of structural designs is discussed.

3.1. Building spatial design definition

For optimization, the representation of a building spatial design is an important factor: it determines the selection of design variables that parametrize the solution of the design problem. To that purpose, within the context of the toolbox, building spatial designs are defined as follows. A building spatial design is a collection of one or more spaces. A space is defined by six variables (ID and other meta data disregarded), which are: three variables for the location (x, y-, and z-coordinates) and three for its dimensions (width w in x-direction, depth d in y-direction, and height h in z-direction). All together, a BSD is a set of cuboidal spaces arranged in an orthogonal grid. This restriction enables a focus of the research on the development of design and optimization methods, without the need to define such methods for special cases introduced by round or skew space boundaries. To focus on feasible solutions, spaces cannot overlap and the minimum z-coordinate of a BSD must be smaller or equal to zero ($z_{min} \leq 0$), where $z = 0$ represents the ground surface.

3.2. Building conformal model

A decomposition of a BSD into sub-parts that is convenient for zoning has already been implemented in the BSO toolbox [3,4]. This decomposition is convenient because—among others—it divides a BSD into sub-volumes that together provide many continuous planes, which follow from the surfaces of each space in the BSD. A zoning procedure can then be defined as a combination of these sub-volumes into zones. Various definitions and principles within this decomposition are used in the zoning methodology that is presented in this paper. The decomposition is stored in the so-called building conformal model, Fig. 2 shows its UML (Unified Modeling Language) class diagram. At the hand of this UML class diagram the decomposition is explained.

In the BSO toolbox, a building spatial design can be decomposed using two different models. The geometry conformal model contains

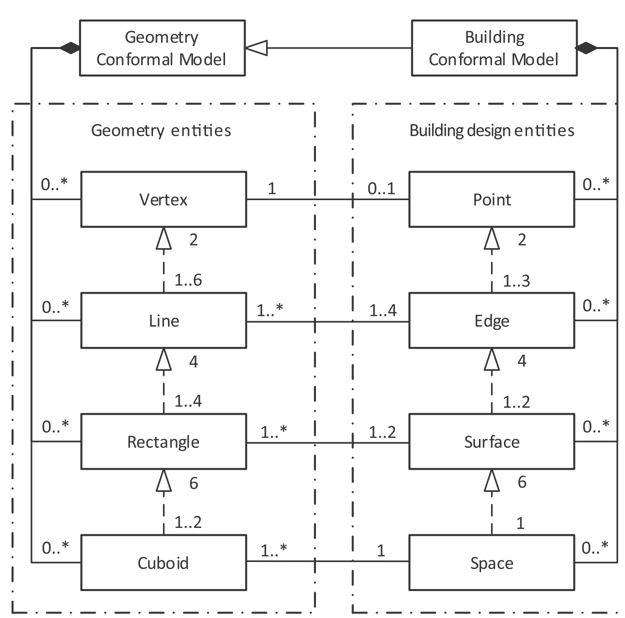

Fig. 2. UML class diagram of the building conformal model.

information regarding the geometry (e.g. connections, volume, and dimensions). Whereas the building conformal model contains information relevant to building spatial design (e.g. what are the surfaces of a space). In the geometry conformal model the BSD is represented by geometry entities: cuboids, rectangles, line segments, and vertices. Cuboids are realized by rectangles; rectangles by line segments; and, rectangles by vertices. The geometry conformal model is initialized such that one entity cannot intersect with another (no intersections) and entities cannot coincide with each other (uniqueness). This is what is referred to as conformal, i.e. the geometries of the entities conform to each other. The second model, the building conformal model, is inherited from the geometry conformal model, creating a new layer of design information. In this new layer, a BSD is represented by the following building design entities: spaces, surfaces, edges, and points. Spaces are realized by surfaces; surfaces by edges; and edges by points. Conformity (i.e. no intersections and uniqueness) is not considered for building design entities. However, the two models are coupled to each other via associations between the different entities. As such, a space is associated with one or more cuboids that are coincident with that space. Vice versa, cuboids are associated with spaces with which they are coinciding. The same rationale applies to surfaces with rectangles and edges with line segments. For more details on the conformal building model, the reader is referred to Refs. [3,4].

The building conformal model is convenient for the generation of domain specific models, e.g. a structural design. In a structural design model, loads and constraints are defined for spatial design features and they are thus assigned to building design entities. However, structural components are assigned to geometry entities, because then the conformal property ensures that adjoining structures are structurally connected in the structural model.

3.3. Structural design models

The toolbox contains a set of tools for structural design generation and evaluation. Such tools are useful when a building spatial design should be assessed for its structural design potential. This section first explains the tool that can be used to evaluate a structural design. Thereafter, a general description of a design grammar is given, which is a tool that can automatically generate a structural design model.

3.3.1. Structural design evaluation

A structural design in the toolbox is represented by a structural model, which is built up from the following components: flat shells, beams, trusses, loads, and constraints. Here, a flat shell (2D) can be loaded with in-plane and out-of-plane forces; A beam (1D) can be loaded in compression, tension, bending, and torsion; And, a truss (1D) can only be loaded in compression and tension. Evaluation of such a model is performed using the Finite Element Method (FEM). To that end, the flat shell, beam, and truss components in the structural model can be meshed (divided) into smaller finite elements. Loads and constraints are then applied to the nodes of the elements, after which the structural response can be analyzed. Using the structural model and the FEM analysis, a structural design can be evaluated, which—in this work—is evaluated for: (i) the structural volume, i.e. the volume of structural components, representing material use, and (ii) the sum of strain energies over all finite elements, which relates to the stiffness of a structural design. For more details on the implementation of the structural model and finite element analysis the reader is referred to Refs. [3,4].

3.3.2. Structural design grammar

A structural design grammar is a set of rules that operates on the building conformal model of the building spatial design in order to create a structural model. Two types of rules are defined here for the structural design grammar. First, a rule that operates on a geometry entity that represents a rectangle. And second, one that operates on a geometry entity that represents a line segment. A rectangle rule is initiated when a rectangle is assigned a structural type by the user or by a user defined process. And, a line segment rule is initiated for line segments that are associated with those rectangles that have been assigned a structural type. Four structural types are defined in the toolbox, `flat shell`, `beam`, `truss`, and `no structure`. Each type, if assigned to a rectangle, will generate a combination of structural components as shown in Fig. 3.

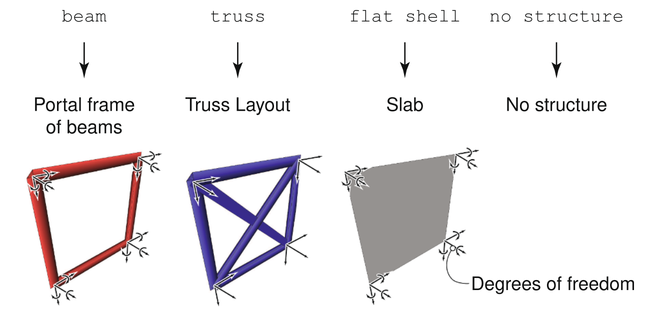

Fig. 3. The different structural types that can be assigned to a rectangle. Note that no boundary conditions are present, these will be added by the line segment rules in a later stage of a design grammar.

If a rectangle is assigned a structural type, the rectangle rule will generate the corresponding combination of structural components shown in Fig. 3 in the structural design model at the location of the rectangle. However, adjoining rectangles with different structural types assigned will cause a conflict in the generation of a component in the adjoining line segment. These conflicts are resolved in the line segments rule set via a hierarchy of components: flat shells have priority over beams, beams over trusses, and trusses over no structure. This can be observed in Fig. 4, where such conflicts occur in the adjoining regions, which are indicated with red line segments.

After a design rule has generated a structure, it is checked if loads or constraints should be applied. Live loads (floor loads) are initialized in the structural model at the location of rectangles that are associated to one or two surfaces (Fig. 2) if the normal vector of these rectangles is vertical within an angle of 45°. Wind loads are initialized at the location of rectangles that are associated with exactly one surface (Fig. 2) and have z-coordinates larger than zero ($z > 0$), which means they are external and above the ground, also see Fig. 5. Three wind load types are specified: pressure, suction, and shear. Each type is assigned in accordance with the direction of the rectangles' normal vector and the direction of the wind load. Finally displacement constraints in x-, y-, and z-direction are initialized at the location of the line segments that are associated with an edge of a space, which are horizontal, and which have a z-coordinate smaller than or equal to zero ($z \leq 0$).

It should be noted that a load can be placed at a location where there is no structure to carry that load. If such a situation occurs, then at that location a flat shell with a low stiffness is initialized (a flat shell because only surface loads are used). This low stiffness flat shell can

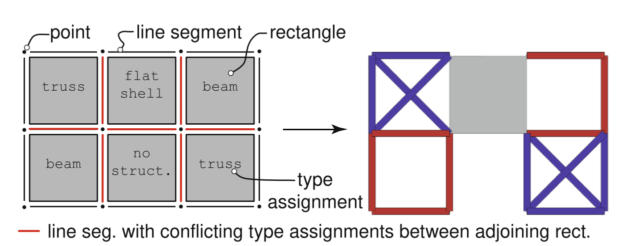

Fig. 4. A two dimensional example of a structural design model created via structural type assignments of rectangles. Conflicting assignments in adjoining regions (red lines) are resolved via a hierarchy of components. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

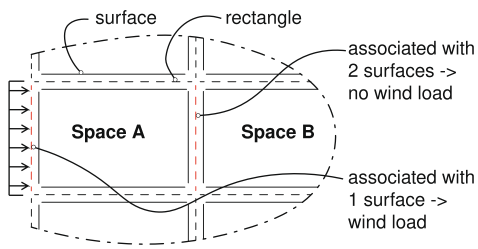

Fig. 5. An example of how the grammar determines if a wind load should be applied. Note that spaces have been shrunk to distinguish between surfaces and rectangles.

transfer the loads to the actual structure without contributing to the stiffness of the structural model. These low stiffness flat shells are only accounted for in the finite element analysis of a structural design, but they are omitted from the visualizations and objective values of a structural design.

4. Spatial zoning

This section explains the developed zoning methodology in two parts. First, in Section 4.1, the concept of zoning in general and zoning specifically for structural design are explained. Second, in Section 4.2, the implementation and the technical details of the zoning procedure in the toolbox are elaborated.

4.1. Concept

Spatial zoning can be seen as a search for individual zones and the process of combining these individual zones into zoned designs. Zones are found by considering combinations of volumes in the building spatial design such that they are meaningful to one or more purposes. For structural design, as discussed in the introduction, the technique of combining whole spaces to find zones can possibly be improved. A volume decomposition of a BSD for zoning should be based on spatial features that are meaningful for structural design. Features that are deemed relevant for structural design are the continuous planes in a BSD in which structural components can be placed. Conveniently, the cuboids in a conformal model provide many continuous planes, which follow from the surfaces of each space in the BSD. In this work, therefore, the zones are defined as a combination of cuboids of a building conformal model (Fig. 2). Accordingly, it is reasoned that zones should be cuboid and as large as possible, because if structural components are then placed at the boundaries of zones, it is likely that these structural components are collocated in continuous planes. Although, it should be noted that the span distances in large zones may be undesirably large, and therefore a notion of a maximum span should also be considered during zoning.

Further explanations of the zoning concept are lead by illustrative examples, which are based on the building spatial design given in Fig. 6. The design has three spaces (solid lines) that are decomposed into five conformal cuboids (dashed lines).

When combinations of cuboids are used, zones can be characterized

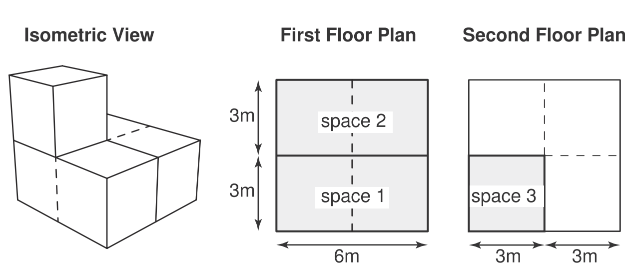

Fig. 6. Two storey building spatial design consisting of three spaces.

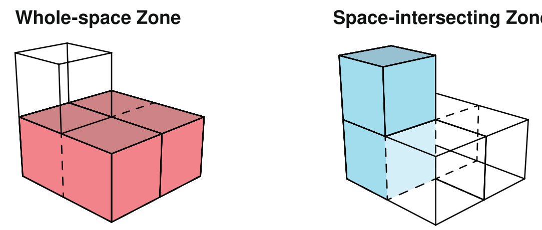

Fig. 7. Two types of zones regarding spatial continuity.

in relation to the spaces: either, as a whole-space zone, of which the surfaces exclusively coincide with the surfaces of spaces; or, as a space-intersecting zone, of which the surfaces may intersect a space. Fig. 7 shows an example of such zones within the design of Fig. 6. In the figure, the BSD is represented by a wire frame and indicated in color are the possible zones: Red (left) for the whole-space zone; And blue (right) for the space-intersecting zone, where the space intersection is indicated with lighter blue.

Spatial zoning does not only consider zones individually, but also combinations of zones. A so-called zoned design consists of a combination of zones, such that it contains all the volumes of a BSD. Combining zones is—in this work—performed by first defining a zoned design for each found zone, containing only that zone. Then, from each zoned design, multiple new zoned designs are generated by adding one other zone (that has not yet been added) to that zoned design. Two constraints are enforced during the addition of zones: First, an addition of a zone cannot result in an overlap of zones; And second, an addition cannot result in a zoned design that has already been found, i.e. each zoned design must be unique. If a constraint is violated, the addition does not result in a new zoned design. Note that the constraint regarding non-overlap does not allow the zones from Fig. 7 to both be present in one zoned design. It is thus possible that a zoned design does not contain all of the volumes within the BSD. Zones are therefore categorized into primary zones and additional zones. Primary zones are the zones that have initially been found. Additional zones are only created after the combination procedure such that they complete the zoned designs that cannot be completed by primary zones alone.

This approach to zoning limits the amount of zones that are initially created and therefore allows to combine them into zoned designs exhaustively. For example in Fig. 7, it can be observed that the red zone (on the left) cannot be combined with the blue zone (on the right) without creating an overlap. An additional zone is therefore created in Fig. 8 (green zone in the middle) to complete the zoned design that was initiated with the red zone (on the left).

An additional constraint is introduced to zones in order to limit the number of possible zones. This constraint disregards any zone that is a subset of another (larger) zone, and as such, only the largest possible zones are found. For the structural design it is assumed that a structural designer has a (pre) conception of a suitable structural system for the building spatial design. This goes along with a notion of the maximum possible span within that structural system. Therefore, a maximum is introduced for the smallest horizontal span of a zone, which may limit the size of the largest possible zones. The influence of defining a maximum span is illustrated in Fig. 9, where, again, the wire frame of

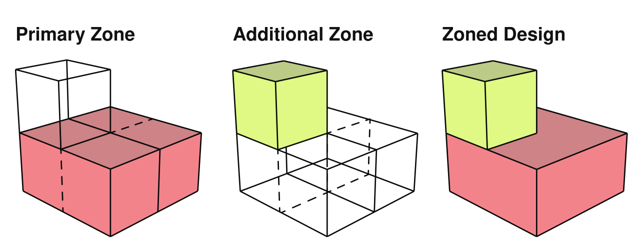

Fig. 8. Completion of an incomplete zoned design by adding an additional zone.

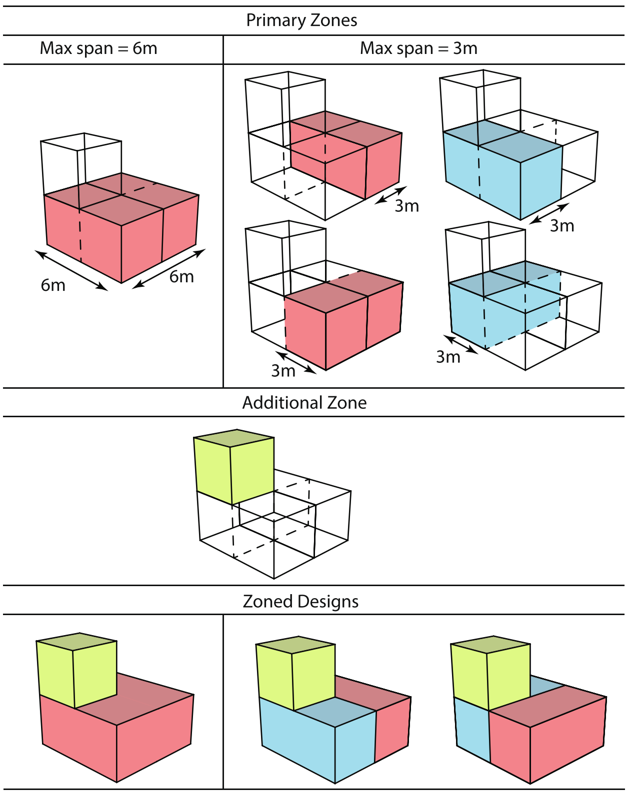

Fig. 9. Different zoned designs result if a zoning is performed for a different maximum span.

the BSD of Fig. 6 is shown and different zones are indicated in color. The figure shows that, with a maximum span of 6m, zoning only results in a single zoned design. Whereas, if the maximum span is set to 3m, zoning results in two zoned designs. The smaller zones that are found using a maximum span of 3m are not found when searching for zones using a maximum span of 6m, because they are a subset of the zone with spans of 6m. Note that, for illustrative purposes, not all possible zones are depicted in Fig. 9, e.g. the space-intersecting zone of Fig. 7 is excluded.

4.2. Zoning procedure

The concept of the zoning procedure, explained in Section 4.1, is further elaborated in this section. First, it is explained how primary zones are found. Accordingly, the combination procedure of primary zones is presented. And finally, the creation of additional zones in the zoning procedure is elaborated.

4.2.1. Creation of primary zones

The spatial zoning procedure is designed to find zones by searching the largest sets of cuboids within the conformal model of a building spatial design. This is achieved in five steps, which are explained at the hand of the building spatial design in Fig. 10.

Step 1: Longest Horizontal Rows of Cuboids. The zoning procedure first finds the longest possible rows of connected cuboids in x- and y-direction, such that each cuboid is used once for each row in each direction. A row of cuboids can only be one cuboid wide in the directions that are perpendicular to the row's direction. This is illustrated in Fig. 11 for the layer of cuboids on the first floor of the BSD in Fig. 10. The four rows that are found in x-direction are highlighted in red, and the four rows that are found in y-direction are highlighted in blue. Moreover, for reference, each row is numbered.

Step 2: Expanding Rows of Cuboids. The second step in the zoning procedure checks if the rows of cuboids that were found in the first step

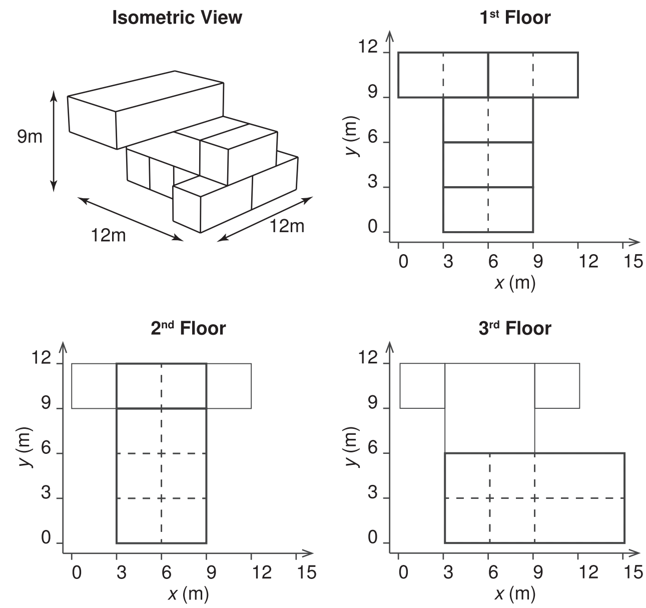

Fig. 10. Isometric view and floor plans of the example building spatial design.

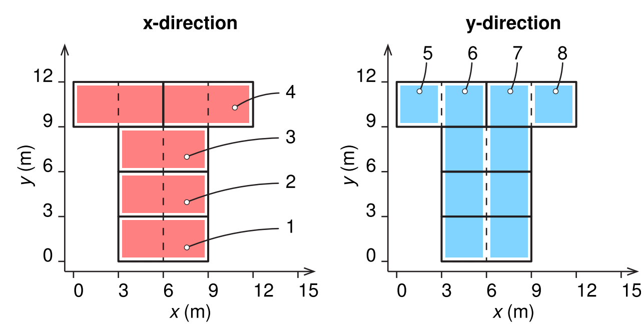

Fig. 11. Rows of horizontal cuboids on the first floor of design 1.

can be expanded horizontally to a larger rectangular shape. To that end, the rows of cuboids are checked for horizontal expansion in the direction perpendicular to the row's direction. For example, for a row in x-direction it is checked if it can be expanded in positive y-direction ($+y$) or in negative y-direction ($-y$). If such an expansion is possible, a zone is initialized and the row of cuboids including the expansion set of cuboids is added (associated) to that zone. Subsequently, the zone is checked again in the same direction for adjacent sets of cuboids that can expand the zone to a larger rectangular shape. This is carried out until no more cuboids can be added on either side of the zone or until an addition would lead to a zone in which the maximum span is exceeded. Fig. 12 shows some examples of row expansions for rows 3, 4, 6, and 8, which were found in Fig. 11 for the first floor plan of the BSD in Fig. 10. Note that expansions of different rows may lead to the same zone, e.g. the expansion of rows 1, 2, 3, 6, and 7 in Fig. 11 all lead to the same zone. Duplicate zones can thus be found, and if so, they are deleted.

Row addition can occur in the two horizontal directions perpendicular to the row. When the row length is larger than the maximum span constraint, it is possible that additions in either direction could lead to the maximum span constraint to be violated. In such cases row addition is not performed, however, if the search is ended for this reason, the zones that are found depend on the initial search direction. Therefore, to resolve this sensitivity, four different search patterns are defined:

1. Search $-x$- or $-y$-direction first, then $+x$- or $+y$-direction alternately.
2. Search $+x$- or $+y$-direction first, then $-x$- or $-y$-direction alternately.

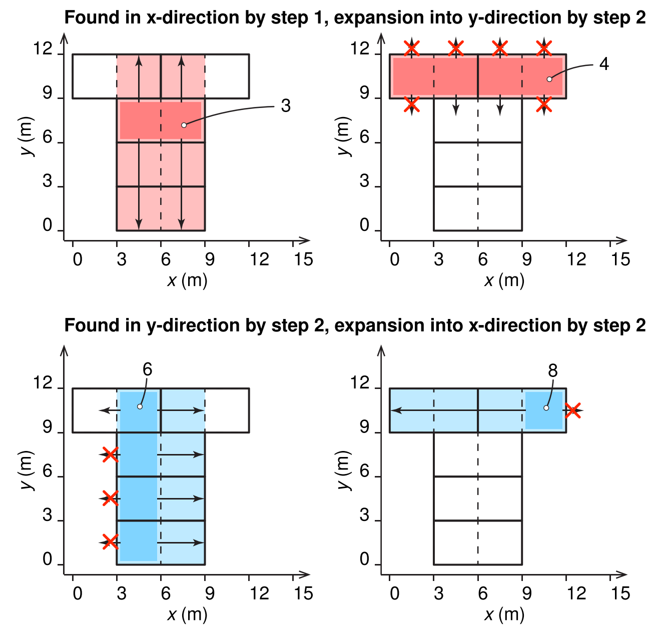

Fig. 12. The second step in the zoning procedure is to expand the rows of cuboids in horizontal direction, perpendicular to the row direction.

3. Search $-x$- or $-y$-direction only.
4. Search $+x$- or $+y$-direction only.

Search pattern 1 is the default, pattern 2 is only employed if pattern 1 is stopped because an addition would violate the maximum span constraint. Moreover, patterns 3 and 4 are employed, if an addition in either pattern 1 or 2 would violate the maximum span constraint. Fig. 13 visualizes the patterns for a maximum span constraint of 12m. In the figure a floor plan is depicted, in which the cuboids (each 3 × 3m)

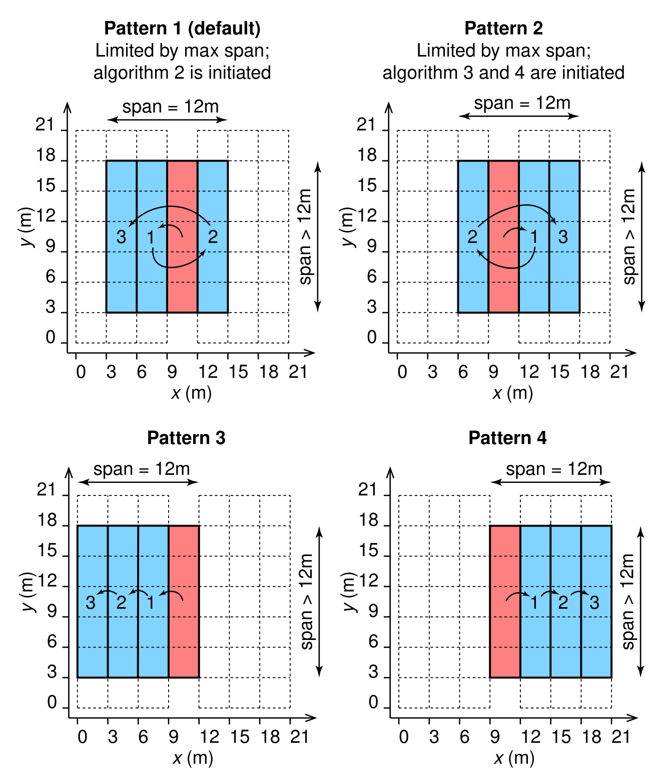

Fig. 13. The different search patterns that can be employed to expand a row of cuboids.

of the building conformal model are indicated with dotted lines, the initial row from which zone addition is started is indicated with a solid line and a red fill, and the row additions with a solid line and a blue fill. Note that the length of the initial row exceeds the maximum span constraint, therefore, in this case, a maximum of three row additions, left or right, can be performed.

When the search for additions has finished and it has resulted in a unique zone, it is checked whether the created zone intersects with spaces. This is carried out by subjecting every cuboid that is associated to the zone to the following check: query the space that is associated to the cuboid and subsequently check every cuboid that is associated to that space for an association to the zone. If one or more of the space's cuboids are not associated to the zone, it is concluded that the zone intersects with a space. If so, the procedure backtracks the steps of the search algorithm for the largest possible zone that does not intersect with any space. If such a zone is found and it is unique, it is added to the primary zones, and marked as a forced whole-space zone. These forced whole-space zones can serve as an alternative for the larger space-intersecting zone. Since, a slightly smaller zone may be preferred in order to avoid space intersecting structural components. An example of how a forced whole-space zone is created from a larger space-intersecting zone is shown in Fig. 14 on the floor plan of the first floor of the BSD in Fig. 10.

Step 3: Upward Zone Expansion. In step 3, it is checked if the zones can be extended upwards. This process starts by iterating the zones from the bottom of the BSD, working upwards. For each zone, the layer of cuboids above it is searched for a collection of connected cuboids with exactly the same rectangular-shaped base at the same x- and y-locations. If such a collection of cuboids is found, the selected zone is expanded by associating the found collection to that zone. If a zone exists that is defined by the exact same cuboids (not more, not less) that were used for the upward expansion, then that zone is deleted. Subsequently, the next cuboid layer above the expanded zone is checked for other cuboids that can expand the zone. As such the procedure is working its way up the building until no new expansions are possible. Upward expansion is illustrated by Fig. 15 for the BSD of design 1 (Fig. 10). In the figure the red zone is checked for expansion upward, such an expansion is possible with the cuboids indicated in blue. Coincidentally, the collection of blue cuboids is also an existing zone, and so this zone is deleted.

The forced whole-space zones are also checked for expansion upwards. However, if an expansion is possible, the procedure first checks if this leads to a space intersection. If so, the expansion is not performed and the search is ended.

Step 4: Overhanging Zones. In step 4, the procedure tries to find zones that can serve as a main structure to support the possible overhanging parts of a BSD. It is possible that such zones are overlooked by the first three steps of the procedure, because these steps look for the largest possible zones. Step 4 is only employed if a zone is found that meets both of the following requirements:

- The minimum of the z-coordinates of the cuboids that are associated

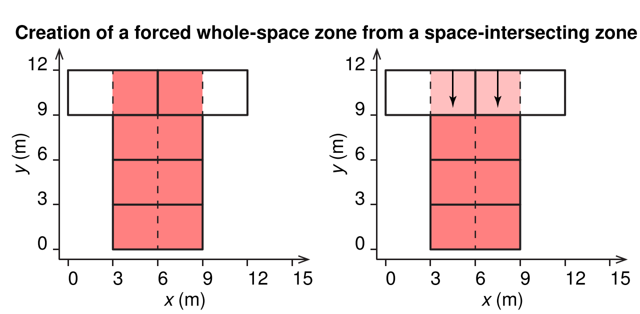

Fig. 14. An expansion of a space-intersecting zone (left) is retracted such that it becomes a whole-space zone (right).

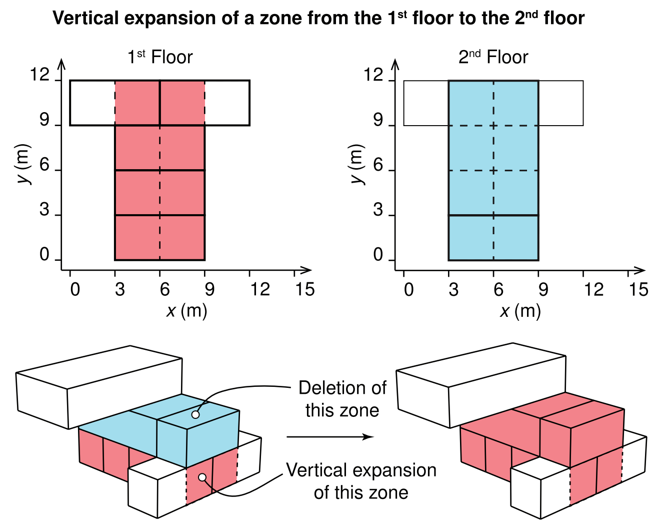

Fig. 15. A zone (red) is expanded upwards. A completely coinciding existing zone (blue) is removed. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

to the zone is larger than zero ($z_{min} > 0$).

- At least one cuboid of the zone is not connected to another cuboid directly beneath it.

If such a zone is found (Fig. 16, left), the procedure selects this zone and searches for sets of cuboids that meet all of the following requirements:

- The set is cuboid-shaped and completely filled with cuboids (cuboids of the geometry conformal design model), i.e. no voids exist within the set.
- The maximum z-coordinate of the set of cuboids is equal to the maximum z-coordinate of the zone.
- The minimum z-coordinate of the set of cuboids is less than the minimum z-coordinate of the zone.
- All cuboids in the set that are located at the same height as the overhanging zone, are part of the overhanging zone.

With the largest set of cuboids that meets these requirements

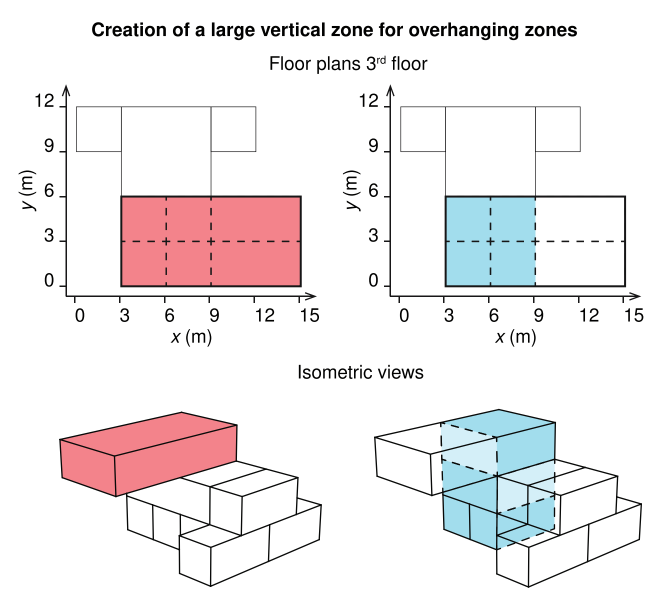

Fig. 16. An overhanging zone (left), and the zone created in step 4 (right) to support the overhanging zone.

(Fig. 16, right) a new zone is created. It should be noted that an overhanging zone and the newly created vertical zone related to it, cannot both be present in the same zoned design because they overlap by definition. If needed, an additional zone is created for the remaining cuboids of the overhanging zone during the combination process.

Step 5: Vertical Cores. In the last step, vertical cores in the building spatial design are detected and stored as a zone. The horizontal dimensions of cores, such as staircases, are likely to be less than the user-defined maximum span. Therefore, the spatial zoning procedure can overlook these zones in the first two steps. However, vertical core structures may contribute significantly to bearing the gravity and/or lateral loading of a building [30]. The fifth step assumes that vertical cores are defined as a single space that spans multiple floors of the building. In this step a zone is created for each space that spans multiple cuboid layers, and has top and bottom surfaces that are external (i.e. associated to exactly one space).

4.2.2. *Combination of zones*

After the described steps to find the primary zones, the zoning procedure will start to combine these zones into so-called zoned designs. A zoned design may not cover all the volume of the BSD, in that case it is called an incomplete zoned design, and if it does, it is called a complete zoned design. Combination starts by initializing a unique zoned design for each primary zone, containing only that zone. Accordingly for each zoned design it is checked whether another primary zone can be added. When a new zone is added, a new zoned design is initialized, which contains the zones of the checked zoned design and the added zone. Subsequently, also the newly created zoned designs are checked for the possible addition of other zones. It should be noted that during the addition of zones, three constraints are considered:

1. The added zone does not overlap with any of the zones that are already present in the zoned design.
2. Adding the zone does not result in a zoned design that already exist.
3. If a zoned design is initialized with a forced whole-space zone, then in the combination procedure only whole-space zones can be added.

The third constraint is considered, because a forced whole-space zone suggests that space-intersecting structural components are avoided, and combining such a zone with space-intersecting zones is contradicting.

A schematic visualization of the combination procedure is given in Fig. 17 using Euler diagrams. On the top of the figure, the Euler diagram of all zones is depicted, showing three primary zones. In the three columns below in the figure, the combination process is illustrated, with in the left column the initialization of zoned designs, and in the other columns all possible combinations. Some combinations result in a constraint violation, and if this is the case, the combination does not result in a new zoned design. The combination procedure ends when all zoned designs (initial + newly found) have been checked.

A user of the zoning procedure is given an option to retain or to remove a zoned design if a zone has successfully been combined with it into a new zoned design. This option provides the user of the zoning procedure with a larger or smaller solution space regarding additional zones, if desired. As an example, temporary zoned designs 1 and 2 in Fig. 17 can be removed using this option, because they both can lead to temporary zoned design 4.

4.2.3. *Additional zones*

After the combination procedure, every zoned design is checked for completeness. This is checked by verifying if all of the cuboids in the building conformal model are associated to a zone. If a cuboid exists that is not associated to a zone, then the zoned design is incomplete. Additional zones should then be created to complete an incomplete zoned design. This is achieved by finding zones in the set of cuboids

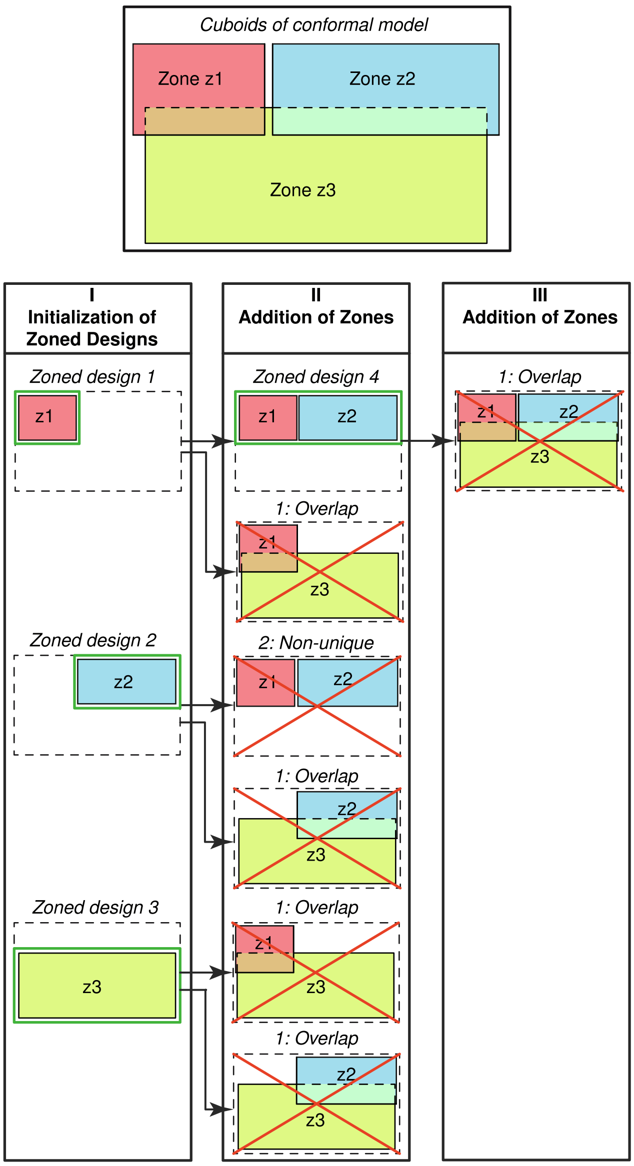

Fig. 17. The combination procedure by Euler diagrams.

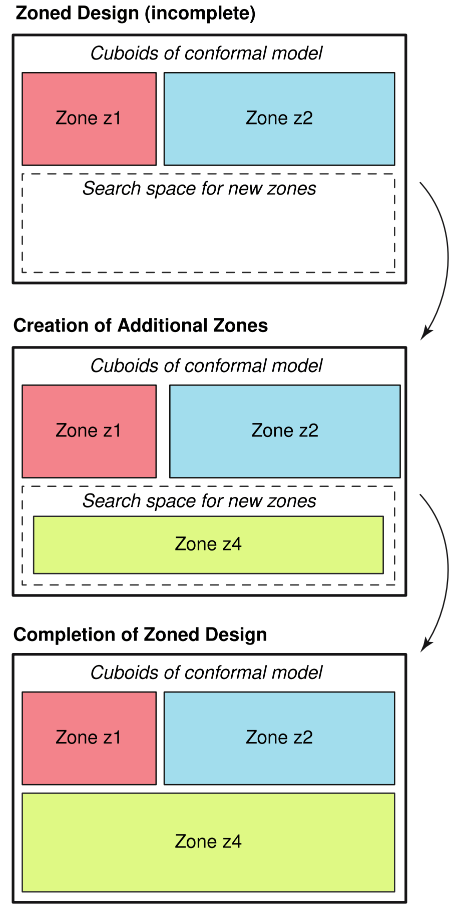

Fig. 18. Completion of a zoned design by finding an additional zone.

that are not associated to the incomplete zoned design in question. Therefore, steps 1, 2, and 3 of the search procedure (4.2.1) for the primary zones are performed again for only that set of cuboids. Fig. 18 illustrates this procedure via the Euler diagram of zoned design 4 from Fig. 17.

After the search for additional zones, the combination procedure is performed again to check if the incomplete zoned designs can be extended with an additional zone. However, removing a zoned design after a zone has successfully been combined with it into a new zoned design is no longer optional, instead it is removed by default. After each combination procedure, the zoned designs are checked for completeness again. If incomplete zoned designs still exist, the search for additional zones is started again, and consequently a combination procedure is initiated. This process is repeated in an iterative fashion, until all zoned designs are complete.

5. Structural design

After the spatial zoning procedure has completed, a structural design can be generated for each zoned design. Two grammars have been implemented in the BSO Toolbox specifically for zoned designs. This section first presents the two new grammars. Thereafter, because one grammar (intendedly) generates unstable structural designs, a stabilization procedure is presented. In an unstable structural design (a part of) the structure is kinematically indeterminate, meaning that it can move freely in one or more directions. This stabilization procedure adds new components to a structural design model to prevent this freedom of movement, until the model becomes stable.

5.1. Structural design grammars

Two structural design grammars have been developed for zoned building spatial designs. The first grammar assigns by default the `beam` type (see Fig. 3) to rectangles that are coincident with the surface of a zone. However, if such a rectangle is also associated with the surface of a space (Fig. 2), then the `flat shell` type is assigned. This grammar will always result in a stable design, and therefore—in this work—it is termed the stable grammar. The stable grammar is intended for zoned designs, however if used on an unzoned design, each space is—in the grammar—considered to be a zone. The second grammar assigns by default the `truss` type to rectangles that are associated to the surface of a space (Fig. 2). However, different from Fig. 3, only a rectangular frame of truss components is generated, i.e. the frame is not braced and thus unstable. Consequently, the second grammar always results in an unstable design, therefore—in this work—it is termed the unstable grammar.

5.2. Stabilization

Stabilization aims to resolve instabilities in a structural design. Smulders and Hofmeyer [31] present a stabilization algorithm, upon which the stabilization procedure described in this section is based, but now including the zoned characteristics of a design. The procedure can be decomposed into four iterating steps: (1) finding vertices in the conformal model that can move freely in the structural design, so-called free-moving-vertices; (2) finding vertices in the conformal model that are located in the proximity of a free-moving-vertex, so-called key-vertices; (3) adding structural components to stabilize a free DOF in-plane; and finally, (4) adding structural components to stabilize a free DOF out-of-plane. Adhering to these steps, hereafter, the stabilization procedure is explained in more detail.

Step 1: find free-moving-vertices. In the finite element model of a structural design, each FE node can have six degrees of freedom (DOFs), three translational and three rotational DOFs. By associating elements to a node these elements may offer resistance against a movement in these DOFs. Hofmeyer and Russel [18] presented a method to find all the DOFs that offer no or very little resistance against movement. Their method is used in this work to generate a list with all FE nodes that can move freely in one or more directions. This list is then used to identify the vertices in the conformal model that—in the structural design—can move freely, such vertices are from here on termed the free-moving-vertices. When each vertex in the building conformal model has been checked for free DOFs and free-moving-vertices were found, the stabilization procedure moves on to step 2. If no free-moving-vertices were found, the stabilization procedure is successful and it is stopped.

Step 2: find key-vertices. In order to stabilize a free-moving-vertex, it is important to find a location to which it can be anchored. Candidate anchor locations are here called key-vertices, for each translational DOF key-vertices are defined (i.e. for DOFs u_x, u_y, and u_z), also see Fig. 19. For each DOF, three sets of key-vertices are selected out of the vertices in the building conformal model, where each set is in-plane with one of the orthogonal planes (i.e. *xz*-, *xy*-, and *yz*-plane) also see Fig. 19. For the two planes with which the DOF's direction is in-plane, the vertices that are diagonally neighboring the considered free-moving-vertex are selected (Fig. 19). For the plane with which the DOF's direction is out-of-plane, the vertices that are orthogonally neighboring the considered free-moving-vertices are selected (Fig. 19). Note that, key-vertices that are co-linear with the DOF are not included, mainly because in the grammar already a truss component has been placed in these directions. Moreover, rotational DOFs (r_x, r_y, r_z) are not considered, because these only exist at the location of beam or flat shell components. If they exist, the beam or flat shell component automatically offers resistance against a free rotation. Once the key-vertices have been listed for each free-moving-vertex, the stabilization procedure moves on to step 3.

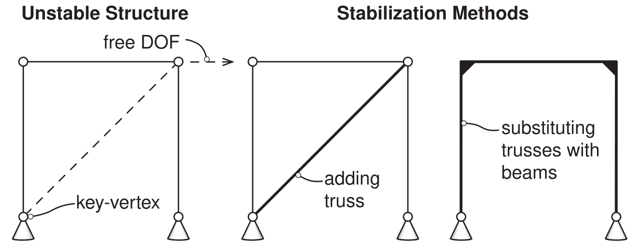

Fig. 20. Two examples of stabilizing structural adjustments for in-plane stabilization.

step 3: place in-plane stabilizing components. Free-moving-vertices are considered for stabilization in ascending geometric order, iterating in *z*-direction first, followed by *y*- and *x*-direction respectively. When a free-moving-vertex is considered for stabilization, its free DOFs are considered for stabilization in the following order: first *x*-, then *y*-, and finally *z*-direction. Stabilization of a free DOF starts by first considering the in-plane key-vertices (Fig. 19). These can be resolved by either adding a truss in the structural model at the location between a key-vertex and the free-moving-vertex, or by replacing the trusses in the structural model that are linking the key-vertex and the free-moving-vertex with beam components. Both methods have been illustrated in Fig. 20. By default the truss addition method is selected, however, if the added bracing would intersect a space, then the second method is selected. The second method is only applied for free DOFs in *x*- and *y*-direction and if the plane in which the key-vertex was found has a vertical orientation (i.e. walls).

The selection of a key-vertex for stabilization is arbitrary, i.e. the order in which they are found. However, a few criteria that may exclude key-vertices from this selection are defined:

1. The key-vertex can move freely in the same direction as the DOF that is considered for stabilization.
2. The key-vertex results in a structural component that intersects with another structural component.
3. The key-vertex and the free-moving-vertex are not in-plane with the surface of a space.

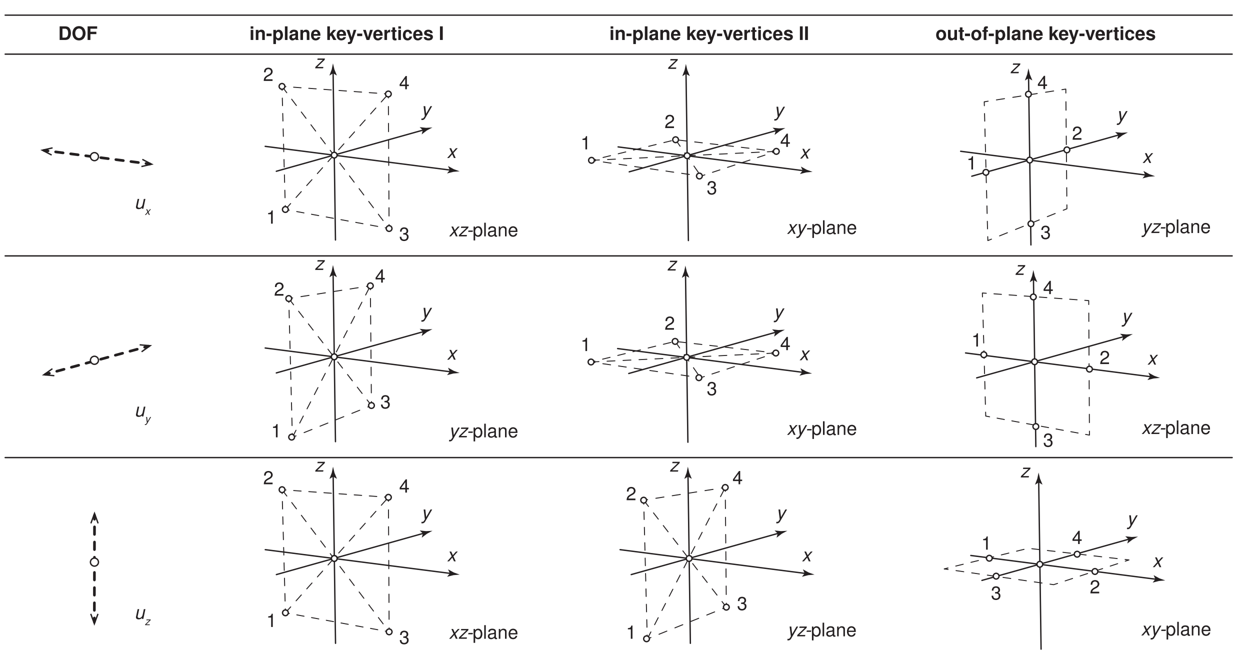

Fig. 19. The key-vertices that are relevant for each free DOF.

4. The key-vertex and the free-moving-vertex are not in-plane with the surface of a zone.

If the BSD has not been zoned, criteria 1, 2, and 3 are active. Whereas if it has been zoned, criteria 1, 2, and 4 are active. If a valid key-vertex has been found, and a stabilizing structural adjustment has been performed, the stabilization procedure will return to step 1. Returning to step 1 ensures that the list of free-moving-vertices is updated to the adjusted structural design, since an adjustment may affect more than one free-moving-vertex. Moreover, due to the exclusion of key-vertices it is possible that for a free DOF no key-vertices remain, and thus it cannot be stabilized. If that is the case, first the other free DOFs of the free-moving-vertex are considered for stabilization. If also that is not possible, or if no other free DOFs are present, the next free-moving-vertex is considered. Finally, when none of the existing free-moving-vertices can be resolved by step 3, the stabilization procedure moves on to step 4.

step 4: place out-of-plane stabilizing components. Out-of-plane stabilization is considered in the case where in-plane stabilization does not yield a stable construction, e.g. due to the exclusion of key-vertices. Selection of free-moving-vertices and their free DOFs is carried out in the same fashion as with the in-plane stabilization procedure. However, this time the out-of-plane key-vertices are listed. Subsequently, each truss component that—in the structural model—is located between the free-moving-vertex and a key-vertex is replaced with a beam component. Fig. 21 illustrates the out-of-plane stabilization procedure. For the same reasons as in step 3, after a stabilizing adjustment has been performed, the procedure returns to step 1. However, when there still exist free DOFs, but no valid structural adjustments can be made, the procedure generates an error. This has however not yet been observed.

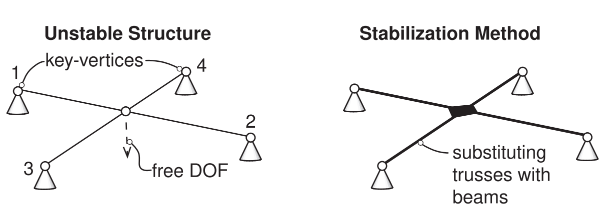

Fig. 21. Example of the stabilizing structural adjustments for out-of-plane stabilization.

6. Case study

In this section, the zoning procedure is used for a case study. The study is applied to three distinct building spatial designs, which are shown in Fig. 22. The designs are selected such that they represent the following archetypes: a small building with an overhang (the building spatial design from Fig. 10); an apartment building with horizontal walkways; and a large production hall with some office spaces. For each building spatial design, several configurations of the settings for the zoning procedure and grammars are tested. The resulting structural designs are compared regarding their structural volume and the strain energy that results from their loading. In the following parts of this section, first the different configurations of settings are discussed, after which the results are presented.

6.1. Settings

Here, the settings that are used for the case study are described. First of all, during each zoning procedure, each zoned design can be assigned any of the following labels: 'L' for large solution space, this label is given to all solutions of the zoning procedure; 'S' for small solution space, this label is given to solutions that result from the zoning procedure if the user selects the option to remove zoned designs after a new zone has successfully been added to it (creating a new zoned design) at least once; 'WS' for whole-space, this label is given to solutions that contain exclusively whole-space zones, i.e. do not contain zones that intersect spaces. Second, different maximum allowable spans were used for each design: Design 1, 6m; Design 2, 6m and 9m; Design 3, 24m. Here, each design is zoned with a maximum span that is deemed suitable for each design. However, Design 2 is zoned twice, once with a span of 6 m and once with 9 m to investigate the sensitivity of the maximum span setting. Third, both the unstable and the stable grammars are used for all of the found zoned designs, and additionally, each grammar is used for the unzoned design once. Fourth, the properties of the structural components that can be added by the grammars are listed in Table 1. Note that the low-stiffness flat shells will be applied at locations in the structural model where a load has been applied, but where no structure is present to carry that load (as explained in Section 3.3). Fifth to mention are the loads that are applied by the grammar, see Table 2. These loads are applied in five load cases, one for live loading, and one for each wind direction ($+x$, $+y$, $-x$, $-y$). And finally, the settings regarding the mesh of the structural model are as follows: each beam component is meshed into 10 beam elements; a flat shell component is meshed into 100 (10 by 10) flat shell elements; and, a truss component is meshed into 1 truss element.

6.2. Results

Fig. 23 shows an example solution using the stable design grammar for Design 3, together with an illustration of the process through which it was obtained. In the top left of the figure the BSD (Design 3) is depicted, where each space is outlined with a solid black line and each surface is indicated by transparent green. In the centre, the cuboids that are present in the conformal design are shown slightly shrunk. At the top right, a zoned design that resulted from the zoning procedure is depicted. Each cuboid is depicted, and all cuboids that belong to a single zone are indicated with the same transparent color and each zone is outlined with a solid red line. Accordingly, at the bottom left, the resulting structural design of the unzoned BSD is given, and at the bottom right, the resulting structural design of the zoned BSD is given. The flat shells that are located on the external walls of the building are made transparent with a dashed border, so that a view into the internal structural system is possible.

In Fig. 23, the unzoned structural design has walls dividing the upper level of the BSD, however, in the large hall beneath no structure is present to transfer the loads to the (constrained) ground surface ($z = 0$). In the zoned structural design, however, the divisions in the upper level are not present, and in the large hall a beam component is present below the load bearing walls, functioning as a column. Comparing the performances of the two solutions to each other, the zoned solution has both lower strain energy (i.e. is stiffer) and has lower structural volume (i.e. less material use).

Fig. 24 illustrates for design 1 the procedure of the unstable design grammar followed by a stabilization procedure. Similar to Fig. 23, at the top the different spatial models are given. Then, at the bottom centre, the unstable design that is created via the unstable design grammar is shown. To the left of the unstable design, the resulting structural design after stabilization using the unzoned solution is given, and on the right, the resulting structural design after stabilization—using the depicted zoned solution—is given.

In Fig. 24, in the unzoned structural design mostly trusses are present, beams that have been assigned are all a result of out-of-plane stabilization. In the zoned structural design more beams are applied now also by in-plane stabilization. The difference in the structural system of the two solutions can seen in the left wall of the partly overhanging space on the top of the building. In the zoned solution, the structural system of that wall continues downward, intersecting with the space below, as such loads on the system can directly be transferred to the foundation. Conversely, in the unzoned solution, the same loads must be transferred to the foundation via a less direct path. Although

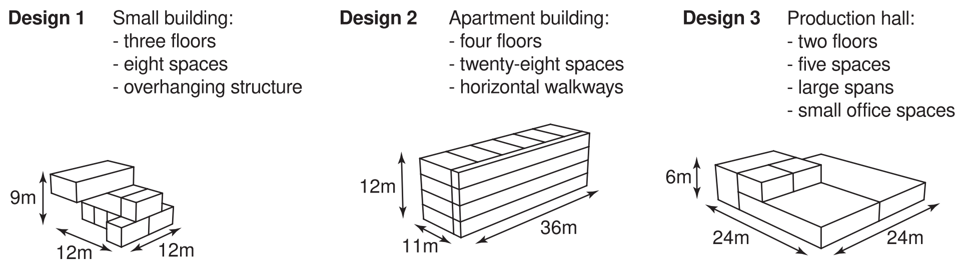

Fig. 22. Isometric views of the four building spatial designs used for the case studies.

Table 1
Structural components and their properties used for the case studies.

Structural component	Dimensions	E [N mm^{-2}]	ν [–]
Flat shell	$t = 150$ [mm]	30000	0.3
Beam	$w \times h = 150 \times 150$ [mm^2]	30000	0.3
Truss	$A = 5000$ [mm^2]	210000	–
Low-stiffness flat shell	$t = 150$ [mm]	0.3	0.3

Table 2
Loads that are applied to the structural models.

Type	Directions	Load [N mm^{-2}]
Live loading on floors	$-z$	0.005
Wind pressure on external surfaces	$+x, +y, -x, -y$	0.001
Wind suction on external surfaces	$+x, +y, -x, -y$	0.0008
Wind shear on external surfaces	$+x, +y, -x, -y$	0.0004

the stabilization adds more structural volume (i.e. material use) to the zoned design, the amount of strain energy in the structure is significantly reduced.

The illustrations in Figs. 23 and 24 show that zoning can have a beneficial effect, however they are typical cases out of all of the possible grammar configurations that have been investigated in the case study. Therefore, to study the effect of zoning more generally, the performance of each resulting structural design in the case study is plotted in Fig. 25. In these plots, each performance is normalized in reference to the unzoned building spatial designs, i.e. $\hat{f}_{i,j} = f_{i,j}/f_{i,unzoned}$ where $f_{i,j}$ is the i^{th} objective value of solution j and $\hat{f}_{i,j}$ is its respective normalized value. The plots in the left of the figures are the results of the stable grammar, whereas on the right the results of the unstable grammar with a consecutive stabilization procedure are plotted. For the stable grammar, the normalized total strain energy and the normalized total structural volume are plotted, whereas for the unstable grammar, the normalized total strain energy and the normalized structural volume that has been added by the stabilization procedure are plotted. Each label (L/S/WS) is plotted with a distinct symbol (square/circle/cross respectively), and additionally the points of the Pareto Front Approximation (PFA) are connected with a red line. Note that solutions labeled with 'S' and 'WS' are a subset of solutions labeled with 'L', and thus in the plot these solutions will always coincide. Moreover, here the PFA is the set of solutions that approximates the Pareto Front, which effectively is the set of mutually non-dominated solutions. A design solution is dominated by another solution if that other solution is performing at least equal in all objectives and at least better in one objective. A design solution is non-dominated if it is not dominated by any other solution in the solution set.

Looking at the results of the stable design grammar in Fig. 25, it is clear that the unzoned solution is always dominated. Which means that zoning has a beneficial effect for at least one of the objectives in all of the case study's building spatial designs. Moreover, for Designs 1 and 3 most of the non-dominated (PFA) solutions are labeled with 'S', which suggests that zoned designs can safely be removed after a new zoned design has been formed by successfully adding a new zone to it. However, a similar trend is not observed for Design 2, which may be caused by the relatively low number of 'S' solutions (2 out of 21). A more comprehensive overview of the results of the stable grammar is given in Table 3. It is clear that the solution sets labeled with 'L' and 'S'

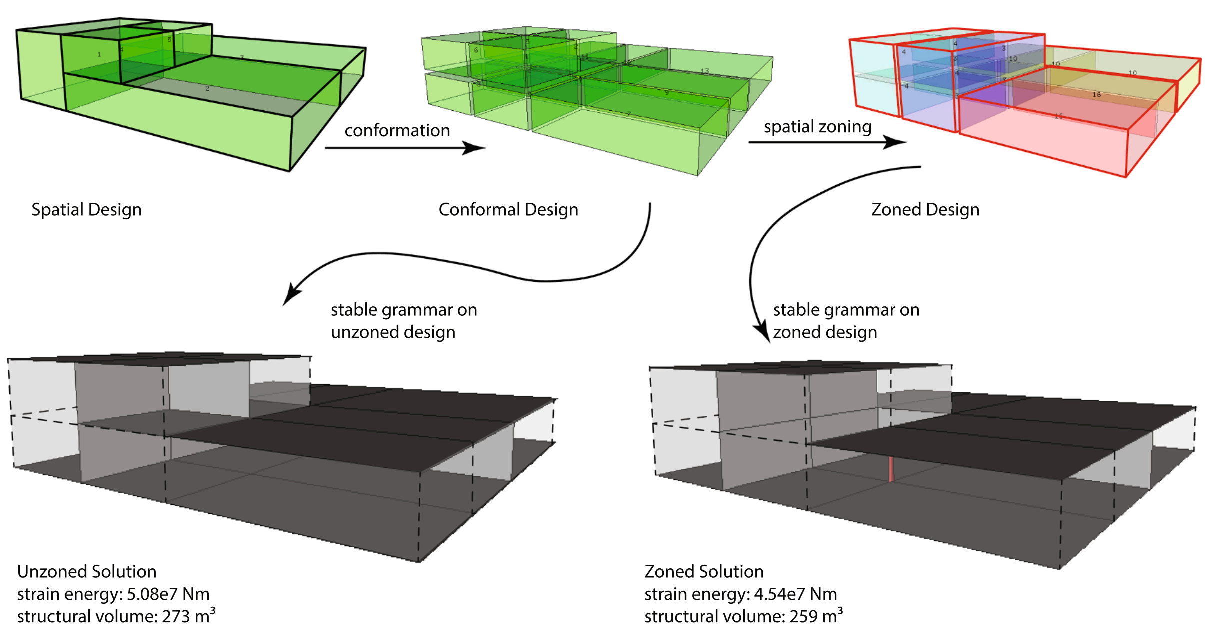

Fig. 23. Procedure of the stable grammar on Design 3 illustrated for both zoned and unzoned BSDs.

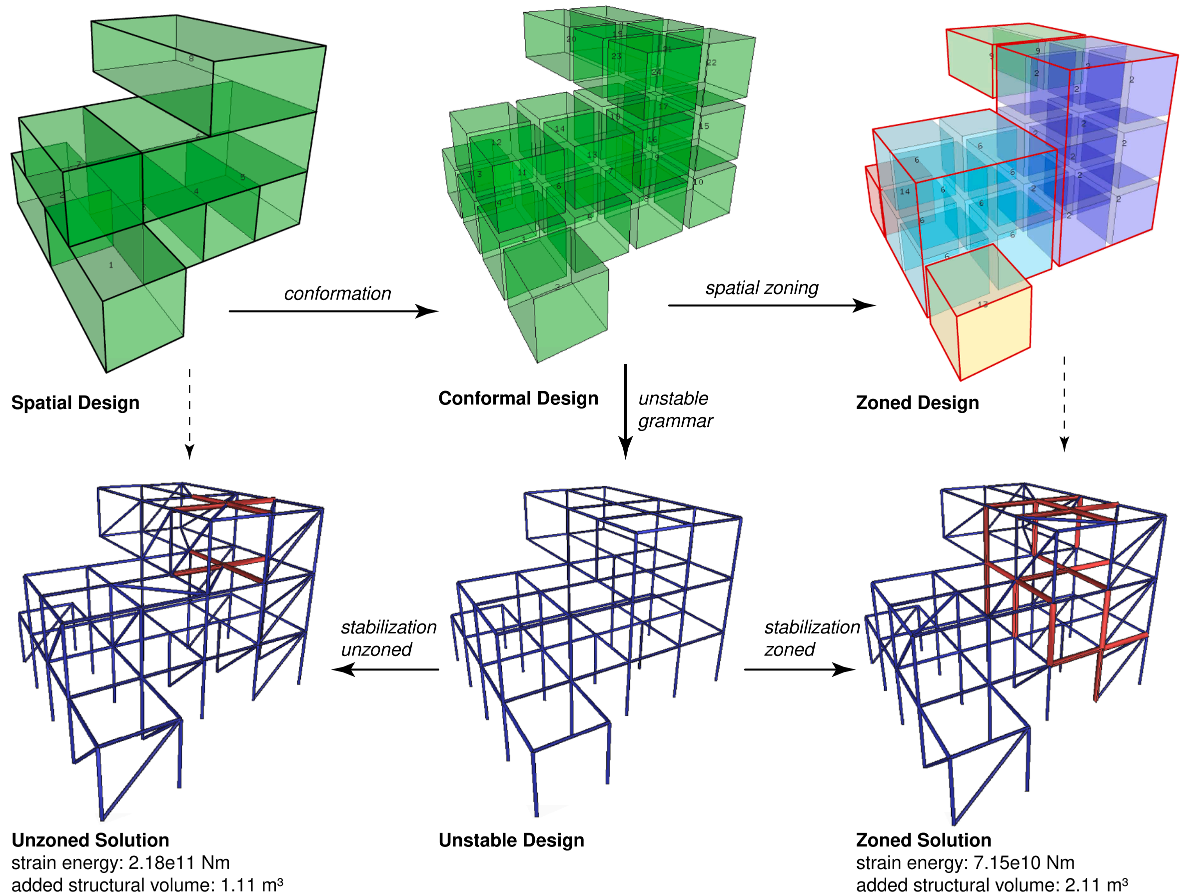

Fig. 24. Design process of the unstable grammar with a consecutive stabilization procedure on design 1 illustrated for both zoned and unzoned BSDs. Note that the purple zone on the complete right results from step 4 of the zoning procedure, which concerns overhanging zones.

contain solutions that are non-dominated. However, the fraction of non-dominated solutions in each labeled set 'L' and 'S' is low. Thus, an arbitrarily selected solution from such sets is likely to be dominated by other solutions. Performing the grammar on just one zoned BSD may thus not result in the best possible structural design: More, and preferably, all zoned BSDs should be evaluated to find the best structural designs for the given BSD. Additionally, in the table it can be observed that solutions labeled with 'WS' are generally dominated by other solutions (in 66% of all cases), however they are not dominated by the unzoned solution. Whole space zoned BSDs thus perform at least similar or better in one or more of the objectives compared to an unzoned BSD.

When focusing on the results of the unstable design grammar followed by a stabilization procedure in Fig. 25, it is noticed that for some BSDs also the unzoned solution is a non-dominated solution (part of the PFA). For Design 2, zoning in conjunction with the stabilization grammar even does not provide a solution that performs better for any objective. Moreover, when applying zoning to Design 2 with a maximum span of 9m, solutions only become worse compared to the unzoned solution. The above suggests that zoning does not necessarily have a beneficial effect in the case where the unstable grammar is used in conjunction with a stabilization procedure. Also for the unstable grammar, in Table 3 a more comprehensive overview is given. From Table 3, it can also be observed that, for the unstable grammar, a fraction of the 'L' and 'S' solutions are non-dominated with the exception of Design 2 with a 9m maximum span. Although, also here this fraction is relatively low, and thus an arbitrarily selected solution from the 'L' and 'S' sets is likely to be dominated by another solution. Regarding the 'WS' solutions, it should be noted that such solutions are sometimes worse that the unzoned solutions.

Finally, with respect to the sensitivity of the maximum span of Design 2, the larger span for the stable grammar shows improvements in the structural volume, however, less improvement and even worse performance in the strain energy of the resulting structural designs. For the unstable design grammar in combination with stabilization, a similar trend is observed for strain energy, but now also for structural volume it is performing worse in the case where the span increases. A sensitivity of the structural volume with respect to the maximum span cannot be observed for the stable grammar, but for the unstable grammar a clear difference can be observed. For the 6m span, 30% extra structural volume is added compared to the unzoned design. Whereas for the 9m span, in some cases up to almost 500% extra structural volume is added compared to the unzoned design. This observation can be explained from the fact that more often an out of plane stabilization needs to be performed when a larger maximum span is allowed. As a consequence, more beams are added, which in the case study have a higher cross sectional area than that of the trusses. The difference in volume between the used structural component thus affects the sensitivity of the structural volume with respect to the maximum span. Regarding the sensitivity of the strain energy with a change in the span, for both design grammars it can be observed that strain energy increases when the span increases. This can directly be related to an increasing span, which leads to larger deformations and thus a higher total strain energy.

7. Discussion

Spatial zoning has in this paper been proposed in combination with automated structural design methods in order to improve such

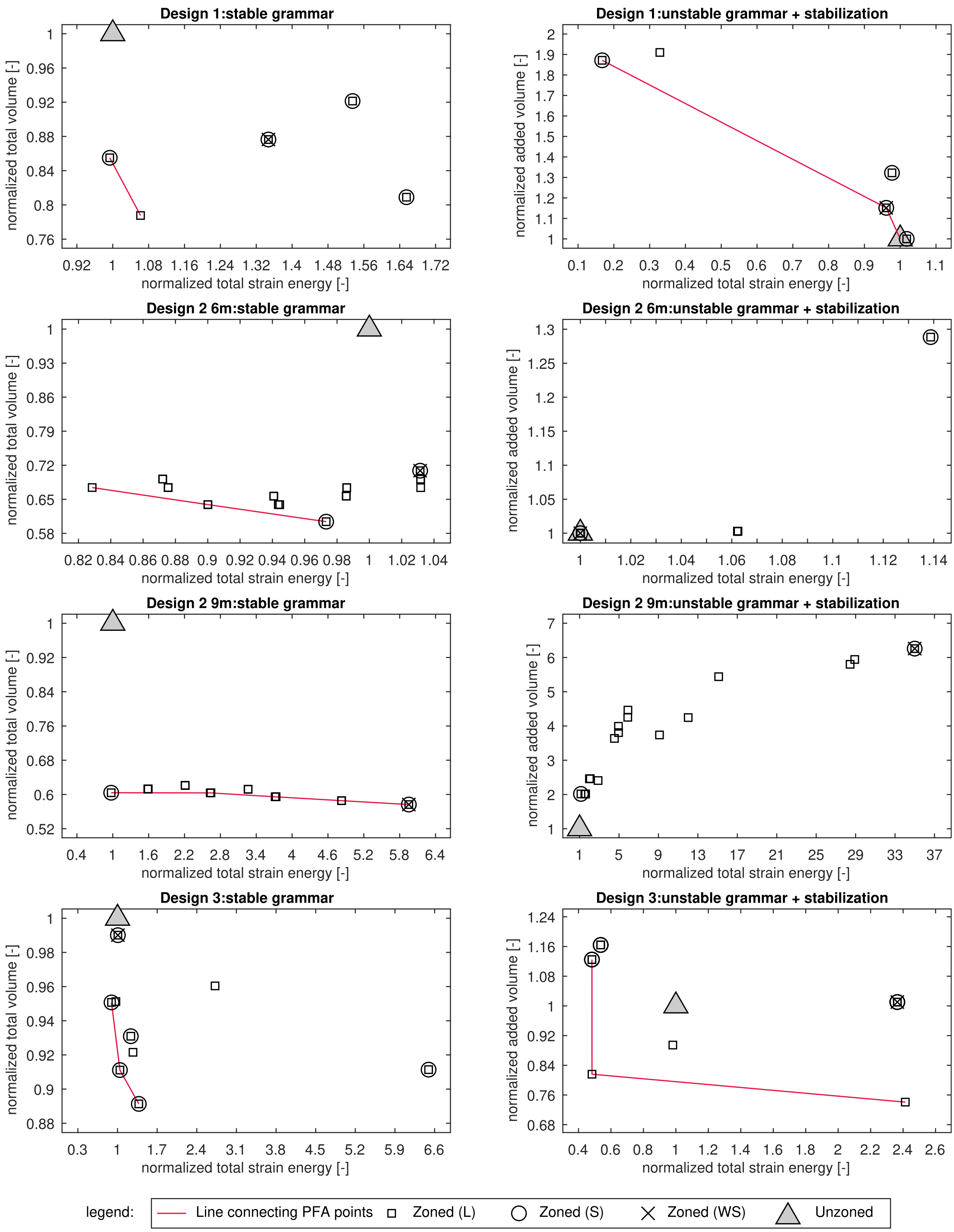

Fig. 25. The normalized results of all case studies, each plotted separately for the stable (l) and unstable grammar (r).

automated methods. Based on studies by Meyer and Fenves [23] and Parent et al. [25] on the structural design process of engineers, a spatial zoning method has been developed such that it finds zones that are promising for structural design. Two automated structural design methods have been defined that take into account the found zones within a building spatial design. One method uses design rules such that the structural model is guaranteed to be stable. Whereas, the other method uses design rules that yield unstable structural models which need to be stabilized with a stabilization algorithm. In a case study, the two automated structural design methods have been applied to unzoned and zoned building spatial designs to investigate the effects of spatial zoning on structural design. Although the results show that zoning has a positive effect on structural design, there are some critical remarks that should be noted.

Table 3
Overview of the results of the stable and the unstable grammar.

	Design 1	Design 2	Design 2	Design 3
Number of spaces	8	28	28	5
Number of cuboids	24	48	48	13
Maximum span	6	6	9	24
Zoning results				
Number of zones (L/S/WS)	14/13/4	27/13/7	36/8/2	17/11/4
Number of zoned designs (L/S/WS)	5/4/1	21/2/1	21/2/1	9/6/1
Results stable grammar				
Non-dominated zoned solutions (L/S/WS)	2/1/0	4/1/0	9/2/1	3/3/0
Unzoned solution in PFA?	no	no	no	no
Fraction of non-dominated 'L' solutions	2/5	4/21	9/21	3/9
Fraction of non-dominated 'S' solutions	1/4	1/2	2/2	3/6
Fraction of 'WS' solutions not dominated by the unzoned solution	1/1	1/1	1/1	1/1
Results unstable grammar				
Non-dominated zoned solutions (L/S/WS)	2/2/1	9/1/1	0/0/0	4/2/0
Unzoned solution in PFA?	yes	yes	yes	no
Fraction of non-dominated 'L' solutions	2/5	9/21	0/21	4/9
Fraction of non-dominated 'S' solutions	2/4	1/2	0/2	2/6
Fraction of 'WS' solutions not dominated by the unzoned solution	1/1	1/1	0/1	0/1

Constraints on Zones. To limit the number of zones, a zone—in this work—has been constrained to cuboid shapes, the largest size possible, and a maximum span. This follows from studies in Refs. [23,25] in which continuous planes—both vertical and horizontal—are found to be of interest for structural design. The largest possible zone ensures that the planes in which structural components are placed are as continuous as possible. To avoid span distances becoming too large, the span distance in one horizontal direction is constrained to a maximum, whereby it is assumed that a designer has a preconception of a suitable span length for the applied structural system. Although the case studies show that the zoning procedure leads to better performing structural designs, it has not been verified if these constraints exclude zoned designs that still result in a structural design that performs well for the objectives. If that is the case, other characteristics than continuous planes, largest zones, and span length may define the structural relevance of spatial zones as well. For instance, in this work, an exception to the largest zone possible is made for vertical cores such as stair cases, however, constraints regarding accessibility (doorways), plumbing (floor/wall savings), or daylighting (windows) that affect the structural design are not considered. Although such constraints can be resolved in later design stages, it would be interesting to investigate if they can already be included by the automated procedure presented here to prevent the need to resolve them. This could, for example, be achieved by limiting the structural types that can be assigned by the design grammars to certain rectangles, e.g. based on spatial or zoning characteristics.

Structural Design Optimality. The applied structural design grammars apply components with fixed dimensions and material properties. It is unlikely that the chosen material properties and dimensions for components lead to an optimal design from the design grammars. It would be interesting to study spatial zoning in conjunction with an automated structural design method that generates more optimal structural designs, for instance by (topology) optimization [2]. Nevertheless, there are indications that these solutions can be useful in a practical design situation; if only by using them to suggest a suitable design [15].

Structural Design Objectives. Structural performance is in this work expressed by two objectives: structural volume and strain energy. Although these two objectives are relevant from a structural point of view, they might be too limited for practical design scenarios. For example, practical objectives like cost, constructability, spatial flexibility and environmental impact have not been considered. Moreover, different materials were used, but the strain energy and volume of both were treated equally, while in practice this is not so straightforward.

Stabilization Procedure. It should be noted that the safety of automatically stabilized solutions has not been assessed. It was pointed out by [31] that the adopted stabilization method stops when a stable design is obtained. As a consequence, when one stabilizing component fails, the design is no longer stable. Therefore, the stabilized designs from the used stabilization procedure are susceptible to progressive collapse.

Building Spatial Designs. For the case study, three different building spatial designs were used, each of which can be characterized by a distinct archetype. Although the case study shows interesting results, a higher number of BSDs is necessary in order to generalize these results and investigate the sensitivities of the parameters. Moreover, spaces and zones are always cuboid within the context of the toolbox. As a consequence all possible BSDs in the toolbox may belong to a subset of BSDs that coincidentally benefit from zoning, whereas BSDs outside of that subset may not (e.g. round spaces). Furthermore, the spatial designs that were subjected to the case study were restricted to be orthogonal and contain cuboid shaped spaces only. In practice, round or skewed spaces may occur as well, which is currently not addressed by the developed method. In order for such building spatial designs to be considered as well, the building conformal model should also be able to decompose building spatial designs in a non-orthogonal context. Additionally, such an adapted building conformal model should also contain volume elements that, together, form continuous planes within the building spatial design.

8. Conclusions and outlook

This paper presents a new method for 3D zoning of building spatial designs, which is tailored to structural design. The method searches for combinations of spaces and sub-parts of spaces that are logical from a structural point of view. Besides the method, two new structural design grammars are presented, which generate a structural design for a building spatial design, thereby taking into account the zones that were found by the new zoning method. One of the design grammars employs a new stabilization method—also presented in this paper—that takes into account a zoned building spatial design.

A case study has been performed on three archetypal building spatial designs using the two new structural design grammars. From the results, it has been observed—for both design grammars—that the structural designs that were generated for a zoned building spatial design generally perform better for the objectives. It is thus concluded that zoning has a beneficial effect on the structural design of a building spatial design.

The zoning method has been tailored to structural design such that it can exhaustively generate zoned designs, i.e. only the zoned designs that are expected to be beneficial for structural design are created. The method has been tailored such that it searches for zones with the following features: (a) a zone is an as large as possible combination of spaces and sub-spaces (allowing space-intersections); (b), the smallest horizontal span in a zone is not larger than a maximum defined distance; and (c), a zone is cuboid. Then, when using the zone boundaries to place structural components, it is likely that continuous planes are formed in which structural components are placed, which is beneficial for structural design. The case study suggests that, when searching for zones with such features, the method indeed only generates the zoned designs that are performing well for structural design objectives.

A critical review of the presented work has been given in the discussion in Section 7. From this discussion an outlook on future work is formulated: (i) The case study should be repeated with another 3D zoning method, which also finds zoned designs that have been excluded in this work. This, to investigate if these excluded zoned designs can still perform well, which may even reveal new features to which the zoning procedure can be tailored. (ii) Other objectives should be investigated to also investigate the relevance of zoning with respect to other aspects and constraints of structural design. (iii) The case study should include more building spatial designs, for this allows for a better generalization of the results. And (iv), current research efforts are aimed at lifting the orthogonality constraint for building spatial designs, and the implications for and the extension of the related zoning method are interesting topics for future research.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This work is part of the TTW-Open Technology Program with project number 13596, which is (partly) financed by the Netherlands Organization for Scientific Research (NWO).

References

[1] B.C. Björk, A conceptual model of spaces, space boundaries and enclosing structures, Automat. Construct. 1 (1992) 193–214, https://doi.org/10.1016/0926-5805(92)90013-A.
[2] S. Boonstra, K. van der Blom, H. Hofmeyer, M.T.M. Emmerich, Conceptual structural system layouts via design response grammars and evolutionary algorithms, Automat. Construct. 116 (2020) 103009, https://doi.org/10.1016/j.autcon.2019.103009.
[3] S. Boonstra, K. van der Blom, H. Hofmeyer, M.T.M. Emmerich, J. van Schijndel, P. de Wilde, Toolbox for super-structured and super-structure free multi-disciplinary building spatial design optimisation, Adv. Eng. Inform. 36 (2018) 86–100, https://doi.org/10.1016/j.aei.2018.01.003.
[4] S. Boonstra, H. Hofmeyer. BSO toolbox v1.0, 2020. https://doi.org/10.5281/zenodo.3823893.
[5] N.C. Brown, C.T. Mueller, Design for structural and energy performance of long span buildings using geometric multi-objective optimization, Energy Build. 127 (2016) 748–761, https://doi.org/10.1016/j.enbuild.2016.05.090.
[6] R. Choudhary, A. Malkawi, P.Y. Papalambros, Analytic target cascading in simulation-based building design, Automat. Construct. 14 (2005) 551–568, https://doi.org/10.1016/j.autcon.2004.11.004.
[7] J.M. Davila Delgado, H. Hofmeyer, Automated generation of structural solutions based on spatial designs, Automat. Construct. 35 (2013) 528–541, https://doi.org/10.1016/j.autcon.2013.06.008.
[8] I.G. Dino, G. Üçoluk, Multiobjective design optimization of building space layout, energy, and daylighting performance, J. Comput. Civil Eng. 31 (2017) 04017025, https://doi.org/10.1061/(ASCE)CP.1943-5487.0000669.
[9] T. Dogan, C. Reinhart, P. Michalatos, Autozoner: an algorithm for automatic thermal zoning of buildings with unknown interior space definitions, J. Build. Performance Simulat. 9 (2016) 176–189, https://doi.org/10.1080/19401493.2015.1006527.
[10] T. Du, M. Turrin, S.C. Jansen, A.A.J.F. van den Dobbelsteen, J. Fang, Gaps and requirements for automatic generation of space layouts with optimised energy performance, Automat. Construct. 116 (2020) 103132, https://doi.org/10.1016/j.autcon.2020.103132.
[11] S.J. Fenves, N.C. Baker, Spatial and functional representation language for structural design, in: Gero, J.S. (Ed.), Expert Systems in Computer-Aided Design, IFIP 5. 2. Elsevier Science, North-Holland, 1987, pp. 511–529. doi:10.1184/R1/6466424.v1.
[12] S.J. Fenves, H. Rivard, N. Gomez, SEED-config: a tool for conceptual structural design in a collaborative building design environment, Artif. Intell. Eng. 14 (2000) 233–247, https://doi.org/10.1016/S0954-1810(00)00018-2.
[13] F. Flager, J. Haymaker, A comparison of multidisciplinary design, analysis and optimization processes in the building construction and aerospace industries, 24th International Conference on Information Technology in Construction, 2007, pp. 625–630.
[14] P. Geyer, Multidisciplinary grammars supporting design optimization of buildings, Res. Eng. Des. 18 (2008) 197–216, https://doi.org/10.1007/s00163-007-0038-6.
[15] P. Geyer, K. Beucke, An integrative approach for using multidisciplinary design optimization in AEC, in, Proceedings of the International Conference on Computing in Civil and Building Engineering, Nottingham, 2010 ISBN: 978-0-87262-915-8.
[16] H. Hofmeyer, M.C.M. Bakker, Spatial to kinematically determined structural transformations, Adv. Eng. Inform. 22 (2008) 393–409, https://doi.org/10.1016/j.aei.2008.04.002.
[17] H. Hofmeyer, J.M. Davila Delgado, Coevolutionary and genetic algorithm based building spatial and structural design, AI EDAM 29 (2015) 351–370, https://doi.org/10.1017/S0890060415000384.
[18] H. Hofmeyer, P. Russel, Interaction between spatial and structural building design: a finite element based program for the analysis of kinematically indeterminable structural topologies, in: CONVR2009, Proceedings of the 9th International Conference on Construction Applications of Virtual Reality, 2009 ISBN: 978-1-74210-145-3.
[19] H. Hofmeyer, H.S. Rutten, H.J. Fijneman, Interaction of spatial and structural design, an automated approach, Des. Stud. 27 (2006) 423–438, https://doi.org/10.1016/j.destud.2005.07.002.
[20] E.M. Hoskins, Design development and description using 3D box geometries, Comput. Aided Des. 11 (1979) 329–336, https://doi.org/10.1016/0010-4485(79)90033-2.
[21] L. Khemlani, A. Timmerman, B. Bennen, Y.E. Kalay, Intelligent representation for computer-aided building design, Automat. Construct. 8 (1998) 49–71, https://doi.org/10.1016/S0926-5805(98)00065-X.
[22] C. Krishnamoorthy, S. Rajeev, Artificial Intelligence and Expert Systems for Engineers, CRC Press, 2018, https://doi.org/10.1201/9781315137773.
[23] S. Meyer, S. Fenves, Structural Design of Tall Buildings. Technical Report EDRC12-56-93, Engineering Design Research Center. Carnegie-Mellon University, USA, 1993.
[24] R. Mora, C. Bédard, H. Rivard, A geometric modelling framework for conceptual structural design from early digital architectural models, Adv. Eng. Inform. 22 (2008) 254–270, https://doi.org/10.1016/j.aei.2007.03.003.
[25] S. Parent, H. Rivard, R. Mora, Acquisition and modeling of conceptual structural design knowledge, in: H. Rivard, H. Melhem, E. Miresco (Eds.), Proceedings of the Joint International Conference on Computing and Decision Making in Civil and Building Engineering, Montréal, Canada, 2006, pp. 3519–3528.
[26] M.Y. Rafiq, I.A. MacLeod, Automatic structural component definition from a spatial geometry model, Eng. Struct. 10 (1988) 37–40, https://doi.org/10.1016/0141-0296(88)90014-4.
[27] C.F. Reinhart, J. Mardaljevic, Z. Rogers, Dynamic daylight performance metrics for sustainable building design, LEUKOS 3 (2006) 7–31, https://doi.org/10.1582/LEUKOS.2006.03.01.001.
[28] H. Rivard, S.J. Fenves, A representation for conceptual design of buildings, J. Comput. Civil Eng. 14 (2000) 151–159, https://doi.org/10.1061/(ASCE)0887-3801(2000)14:3(151).
[29] R. Sacks, A. Warszawski, A project model for an automated building system: Design and planning phases, Automat. Construct. 7 (1997) 21–34, https://doi.org/10.1016/S0926-5805(97)00034-4.
[30] B.S. Smith, A. Coull, Tall Building Structures: Analysis and Design, John Wiley & Sons, Inc., 1991 ISBN: 978-0-471-51237-0.
[31] C.D.J. Smulders, H. Hofmeyer, An automated stabilisation method for spatial to structural design transformations, Adv. Eng. Inform. 26 (2012) 691–704, https://doi.org/10.1016/j.aei.2012.03.002.
[32] B. Steiner, E. Mousavian, F. Mehdizadeh Saradj, M. Wimmer, P. Musialski, Integrated structural–architectural design for interactive planning, Comput. Graphics Forum 36 (2017) 80–94, https://doi.org/10.1111/cgf.12996.
[33] G. Suter, F. Petrushevski, M. Šipetić, Operations on network-based space layouts for modeling multiple space views of buildings, Adv. Eng. Inform. 28 (2014) 395–411, https://doi.org/10.1016/j.aei.2014.06.004.
[34] M. Turrin, P. von Buelow, R. Stouffs, Design explorations of performance driven geometry in architectural design using parametric modeling and genetic algorithms, Adv. Eng. Inform. 25 (2011) 656–675, https://doi.org/10.1016/j.aei.2011.07.009.
[35] L. Wang, W. Shen, H. Xie, J. Neelamkavil, A. Pardasani, Collaborative conceptual design—state of the art and future trends, Comput. Aided Des. 34 (2002) 981–996, https://doi.org/10.1016/S0010-4485(01)00157-9.