A Shape Grammar for Space Architecture – I. Pressurized Membranes

Val Stavrev*
Aeromedia, Sofia, Bulgaria

The paper presents a study on the morphology of inflatable membranes and their application in space habitat design. The study is focused on the role of physical constraints on pneumatic membranes as a form generating factor in two aspects (1) the definition of elements (primitive shapes) and (2) the definition of a set of rules. The primitive (generic) shapes for pneumatics must be in an initial equilibrium state. Said condition applies restraint on pneumatic shape primitives. We used geometric surfaces having closed analytic solutions satisfying equilibrium state under uniform internal pressure. They are all surfaces of revolution – sphere, torus, cone, cylinder, and ellipsoid. A special class of intermediate elements – compression rings and tensioned diaphragms – are introduced to support concatenation rules.

Nomenclature

n_ϕ	=	circumferential tension
n_θ	=	meridional tension
$n_{\phi p}$	=	pressure induced circumferential tension
$n_{\theta p}$	=	pressure induced meridional tension
i	=	index of shape in assemblage
p	=	uniform internal pressure
p_R	=	normal component of pressure
R_1	=	meridional radius of curvature
R_2	=	circumferential radius of curvature
ϕ	=	circumferential coordinate
θ	=	meridional coordinate
V	=	set of nonterminal entities
Σ	=	set of terminal entities
V	=	set of production rules
I	=	initial object (axiom)
χ	=	integration variable

I. Introduction

COMPOSING new shapes from a set of generic elements is among the favorite techniques of architects. Behind classical principles of composition new systematic concepts have been examined during the 20th century: Kit-of-Parts (M.van der Rohe), Generative Design and Shape grammars (Stiny and Gips) among others.

The *shape grammars* have been defined in architectural design in a seminal article by George Stiny and James Gips in 1971 as: "A method of shape generation using grammars which take shape as primitive and have shape specific rules". The shape grammars approach is based on *generative grammars* introduced in theoretical linguistics by Noam Chomsky. The idea of generative grammar is that all possible expressions in a particular formal language can be produced by applying in all possible ways the set of replacement rules given by the grammar.

Motivation. The origin of architectural morphology has been largely debated in architectural theory. Here, we will consider an aspect of this discussion not quite often addressed – the impact of physical constraints on morphology. The discussion will be limited to morphology of pressurized envelopes of space habitats. Schemata of

* Membrane structure designer, Aeromedia, Sofia 1000, Bulgaria, nonmember

terrestrial architecture might be inappropriate for architecture of entirely new environments of outer space and celestial bodies. What is definitively distinguishing space habitable modules from virtually all terrestrial architecture is the fact that the space habitat is a *pressure vessel*. The following study is a part of the author's trials toward finding an adequate tool for space habitat envelopes design in early stages of design process

II. Literature and Related Works

The Generative grammars were originally conceived by N. Chomsky as a way to describe natural languages. That promise has not been fulfilled. However its recursively defined concepts in Computer Science have found multiple other applications – music, urban analysis etc.

A. Shape Grammar

Shape grammars were introduced by Stiny [1, 2] in the 70's as a formal approach to architectural design. The shape grammars were successfully used for the construction and analysis of architectural design [3 - 6].

Shape grammars are a production system created by "taking a sample of the whole for which one is trying to write a language" [7]. From this sample a vocabulary of shapes can be written that represents all the basic forms of that sample. By defining the spatial relationships between those forms and how the forms are related to each other, shape rules can be notated. In their original form the shape grammars are based on arbitrary configurations of lines, and therefore hard to handle by computers. In order to alleviate this, Set Grammars will be used [7].

Two types of use of shape grammars may be identified. Analytic shape grammars understand a given design style through decomposition. Generative shape grammars produce new designs by encoding design constraints into rules.

Recently, Generative Parametric Design was successfully used in creation of content for virtual worlds in computer games or movies [8]. The systems of Procedural Modeling of virtual buildings is proposed by Müller et al. [8] using grammars to produce variations of building designs, generated through random or user selected parameter adjustment.

A significant step toward implementation of real world constraints in procedural modeling has been made by Whiting et al. in their recent work [9], where the structural feasibility check is implemented into procedural modeling of buildings. They use an energy function as a measure of infeasibility, and apply gradient-based optimization to select rule parameters that satisfy structural stability constraints. While existing structural analysis tools focus heavily on providing a stress state analysis, the proposed method automatically tunes a set of designated free parameters to obtain forms that are structurally sound. Users of the system may not have intuition about the mechanics that govern structural stability, or knowledge of traditional proportions used in building design.

B. Space Habitat Pressurized Shells

Space architecture is a loosely defined emerging branch. Architects and architectural approaches are involved in interior design of space stations probably since Skylab. Nevertheless, their role in outline design of existing space habitat modules is marginal.

The outer form of modules is dictated by the size and shape of the launch vehicle and the need to contain an atmosphere, but expandable (inflatable) modules should not strictly follow the outline of the launch vehicle cargo bay. The morphology of space habitat pressurized shells may be categorized in two groups. The first group uses generic shapes for the artificial atmosphere envelope – sphere, torus, cylinder with end caps and rarely a cone, as well as other shapes that are "inflatable-like", i.e. the morphology of pressure vessels and inflatables. The second group tries to modify the bubbles of pneumatics into forms that are more familiar to architects – nearly flat walls, orthogonal or hexagonal nesting.

All the practically developed inflatable space habitats fall within the first group - TransHab and the Bigelow Aerospace modules derived from it. Widely accepted as de facto standard in expandable/inflatable space habitats, they comprise a rigid structural core and multilayer inflatable envelope. Bigelow modules reproduce the most common shape of pressure vessel – cylindrical body with hemispherical end caps. TransHab shares its morphology with the automotive tire; even the webbing pattern is made in the same manner as radial tire cord reinforcement. A spherical lunar habitat implemented in different sizes and of different materials has been proposed in a scope of works ranging from the pioneering work of von Braun [10] to recent feasibility studies [11, 12].

An example of the second group of inflatable habitat designs is the "prismoidal" inflatable modular structure demonstrating orthogonal nesting, and shaped in a non-trivial way by applying a high pressure tubular pneumatic frame [13]. A "bubble cluster" is used in an early design made in SICSA. In an ESA sponsored study, rigid elements are used to modify an inflatable rotational ellipsoid.

Many architects and designers explore out-of-atmosphere architecture. These projects are preliminary or visionary designs meant to be inflatable. In my search of literature I found the most nontrivial yet structurally feasible inflatables in the works of Paul Jungmann[14] long before the CAD revolution. This was probably due to the fact that Jungman used real inflatables to create his designs. The CAD systems and especially 3D model tools are almost entirely devoid of physical properties of models created.

C. Pressure Vessel

The term *pressure vessel* is self-descriptive enough but we list different kinds of pressure vessels here:

- industrial pressure vessels – boilers, chemical reactors, compressed gas storage facilities
- pressurized aircraft and spacecraft cabins
- submarine pressure hulls
- super-pressure balloons and dirigibles
- Rocket and launch vehicle propellant tanks
- EVA spacesuits

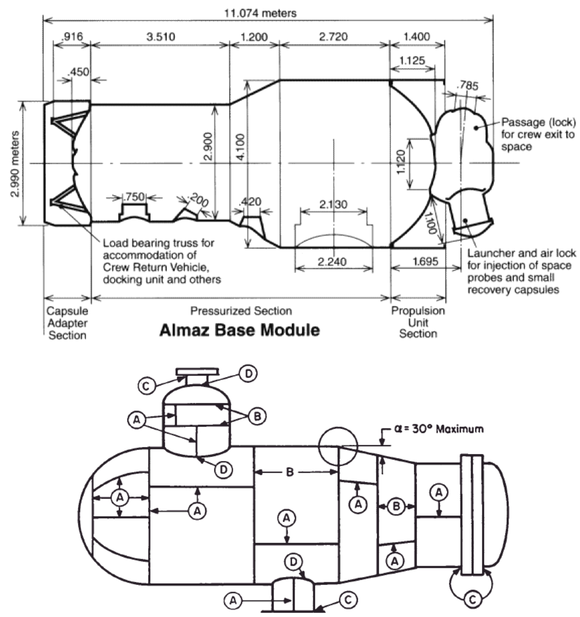

Figure 1. Similarity in morphology of pressure vessels having different typology: typical space module (Russian Almaz) and typical industrial

Pressure vessels and pressure stabilized shell structures are an essential part of aerospace systems and have been studied comprehensively [15, 16].

There are huge quantities of manuals, handbooks, and design codes on industrial pressure vessels – boilers, tanks, and reactors[17, 18]. The most relevant information on aerospace pressure vessels may be found in specialized literature. Recently, non-axisymmetrical pressure vessels have been investigated for future blended body aircraft. Structural analysis of highly pressurized membranes of inflatable space habits may be found there [19].

III. Approach

The main concept of grammar design as used for example in architecture is based on a shape grammar utilizing a rule-driven procedure. Starting from initial axiom shape, rules are applied to replace shapes by other shapes. The rule has a labeled shape on the left hand side, called the “predecessor,” and one or multiple shapes and commands on the right hand side, called the “successor.”

A. Definitions

Production systems are defined for specific objects. For example, strings in Chomski’s grammars or shapes in shape grammars. The set of objects on which a specific system operates is called the vocabulary U. The vocabulary consists of *terminal* symbols $\Sigma \subseteq U$ and *nonterminal* symbols $V \subseteq U$.

A production system contains a list of productions *R* in the form: $v \to u$, $R \subseteq V \times U^{+}$. Objects are produced by starting with an initial object I, and repeatedly applying productions. This is called the production process. The initial object is an arbitrary combination of objects from the vocabulary, and it is called the axiom. The tuple (V, Σ, R, I) is called a generative grammar or simply grammar.

Shape grammars introduce transformation productions on sub-shapes v. Productions can be applied when some transformation $f(v)$ of v occurs in the current object w, formally $f(v) \le f(w)$. The result of an application is the replacement of $f(v)$ with $f(u)$ in the current object, formally $[w - f(v)] + f(u)$. Possible transformations *f* are dependent on the object type, for example, when shapes are used, *f* can be a scaling transformation.

Stiny mentions the transformations: translation, rotation, reflection, scale, and any composite transformation between two of them. The list can be extended or shortened for given applications.

The set of entities generated by application of grammar rules are denoted as ‘words’ in formal languages. In pressure vessel design the composition of sections is usually denoted as ”stack.” We will denote the “assemblages” set of shapes.

”Coalescence” will be used to denote a union of entities.

B. Overview

The proposed approach selects rule parameters by determining values that will make the model stand in initial equilibrium configuration

The structures were modeled as an assemblage of generic shapes and explicit analysis of the force distributions at the interfaces between adjacent elements was substituted by geometric parameter search or conditional rules. Since geometry and stress distribution are strongly coupled in the membrane model used in this study the initial equilibrium can be found by choosing parameters of concatenated shapes, and by adding compression or auxiliary tension elements where needed.

1. Rule Extraction

The procedural rules in said examples may be derived from informal rules of architectonic composition theory and praxis by formalization refining it to algorithmically interpretable level. However, the pressurized membranes are a relatively new and unfamiliar building system and generally the pressure vessel design is not a part of the architect’s compendium. On the other hand, industrial pressure vessel design is a mature branch of structural engineering and like civil engineering it is codified in “design by rules”.

There are guidelines for pressure vessel design according to Bednar [17]:

1. All external loads must be applied in such a way that the internal stress reactions are produced in the plane of the shell only. Membrane stress analysis assumes that the basic shell resistance forces are tension and that a membrane cannot respond with bending or transverse shear forces.
2. Any boundary reactions, such as those at supports, must be located in the meridional tangent plane, otherwise transverse shear and bending stresses develop in the shell boundary region.
3. The shell including the boundary zone must be free to deflect under the action of the stress resultants. Any constraints cause bending and transverse shear stresses in the shell.
4. The change of meridional curve is slow and without cusps or sharp bends. Otherwise bending and transverse shear stresses will be included at such gross geometrical discontinuities.
5. The membrane stress resultants are assumed uniformly distributed across the wall thickness. This can be assumed if the ratio of the radius of curvature [R] to the wall thickness [t] is about $R/t \geq 10$ and the change in the wall thicknesses, if any, is very gradual.
6. The radial stress σ_r is small and can be neglected. A plane state of stress is assumed.
7. The middle surface of the entire shell is assumed to be continuous from one section of the shell component to another across any discontinuity. At the junction the lines of action of the meridional stress resultant N+ are not collinear and this eccentricity introduces additional stresses.
8. The loadings are such that the shell deflections [ΔR] are small ($\Delta R \leq t/2$) and in the elastic range.

To summarize, a shell will carry the load by membrane stresses only if it is thin, properly shaped, and correctly supported.

2. Momentless Shell

Linear membrane theory is the limiting case corresponding to a zero-order approximation, or momentless state. Thin shells, in general, display large stresses and deflections when subjected to relatively small bending moments. Therefore, in the design of thin shells, the condition of bending stresses is minimized or totally avoided. If, in the equilibrium equations of such shells all moment expressions are neglected, the resulting shell theory is called membrane theory, and the stressed state is referred to as a momentless state of stress and such shells are referred to as momentless shells [20, 21].

The membrane stress is considered primary for mechanical loads. The vessel geometries can be broadly divided into plate- and shell-type configurations. The plate-type construction used in flat covers (e.g. airlocks) resists pressure by bending, while the shell-type membrane action operates identically to what happens in membrane under pressure. Generally speaking the shell-type construction is the preferred form because it requires less thickness (as can be demonstrated analytically) and therefore is more structurally efficient no mater what material is used for its manufacture – flexible membrane or rigid metal. Shell-type pressure components resist pressure primarily by membrane action.

There are two types of shells that comply with this membrane theory:

1. shells sufficiently flexible so that they are physically incapable of resisting bending, and
2. shells that are flexurally stiff but loaded and supported in a manner that avoids the introduction of bending strains.

The membrane hypothesis produces the simplest and most readily solvable system of shell equations. If the wall of the shell is thin and there are no abrupt changes in thickness, slope, or curvature and if the loading is uniformly distributed or smoothly varying and symmetric, the bending responses can be very small and negligible.

The equilibrium condition of uniformly pressurized membrane is expressed as [22]:

$$\nabla \overline{N} + p = 0 \qquad (1)$$

where $\overline{N}$ is the membrane tensor and p is the pressure.

C. Geometry of Surface of Revolution

A surface of revolution is generated by the rotation of a plane curve about an axis in its plane. This generating curve is called a meridian. The intersections of the generated surface with planes perpendicular to the axis of rotation are parallel circles and are called parallels. For such surfaces, the lines of curvature are its meridians and parallels. A convenient selection of surface coordinates is the curvilinear coordinate system θ and ϕ, where θ is the angle between the normal to the surface and the axis of rotation and ϕ is the angle determining the position of a point on the corresponding parallel, with reference to some datum meridian.

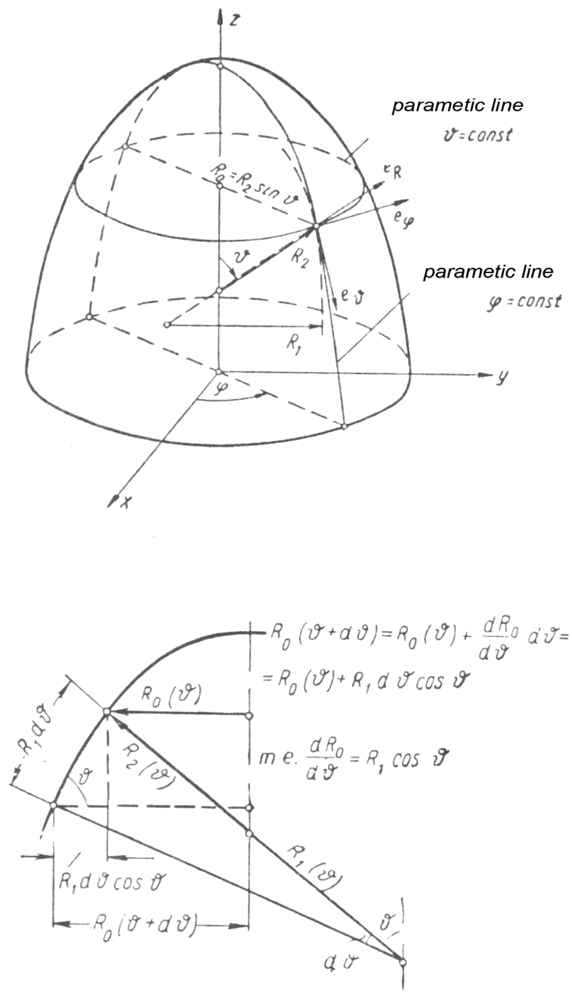

Figure 2. Geometry of shell of revolution. Redrawn from Otto.[22]

Fig. 2 shows a meridian of a surface of revolution. Let R_0 be the distance of one of its points normal to the axis of rotation and R_1 its radius of curvature. In future equations, we will also need the length R_2, measured on a normal to the meridian between its intersection with the axis of rotation and the shell surface. Noting that $R_0 = R_2 \sin \theta$, the surface of the shell of revolution is completely described by R_1 and R_2 which are functions of only one of the curvilinear coordinates, θ. R_0 will be the radius of curvature when $\theta = \pi/2$.

In the generating procedure, we use a set of basic shapes that certainly are in initial equilibrium under uniform internal pressure. There is a class of surfaces of revolution relevant to this criterion and widely used in pressure vessel design. Most common designs include sphere, cylinder, cone, ellipsoids, and torus section.

D. Procedure-Driven Pressure Vessel Design

Stages of shape grammar development are usually: shapes > spatial relations > rules > shape grammar > designs

The production process follows the sequence described in Müller et al.[8] A configuration is a finite set of basic shapes. The production process can start with an arbitrary configuration of shapes *I*, called the axiom, and proceeds as follows:

1. Select an active shape with symbol *B* in the set.
2. Choose a production rule with *B* on the left hand side to compute a successor for *B*, a new set of shapes B_{new}.
3. Mark the shape *B* as inactive and add the shapes B_{new} to the configuration and continue with step (1).

"Active shape" or "marked shape" are actually nonterminal symbols in shape grammar implementations.

When the configuration contains no more nonterminals, the production process terminates.

Production rules are defined in the following form:

id: predecessor : cond => successor : prob

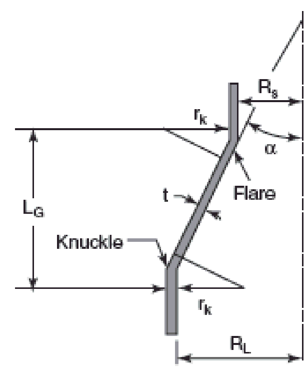

Figure 3. Transitions between two sections of pressure vessel stack: *knuckle* and *flair*.

where *id* is a unique identifier for the rule, *predecessor* is a symbol identifying a shape that is to be replaced by successor, and *cond* is a guard (logical expression) that has to be evaluated in order to apply the rule. The rule is selected with probability *prob*. While condition *cond* contains a left hand shape as argument, such defined rules are case sensitive.

For example, the rule:

1: S_i $S_{(i+1)}$: knuckle => S_i Dph S_{i+1}

replaces the shapes **S_i S_{i+1}** by three shapes **S_i Dph S_{i+1}** or in other words, inserts diaphragm **Dph** if a knuckle has occurred in the joint between **S_i** and **S_{i+1}**.

Since the shell of revolution is defined by its meridional curve, the form-finding procedure is reduced to finding the meridional curve. Thus, the problem is dimensionally reduced from 3D to 2D. The membrane state imposes constraints on the minimum radius of meridional of curvature [3] described by Eq. (2) below. If generic shapes are limited to sphere, cylinder, cone, and torus sectors, the meridional curve will be a polyline – i.e., a curve composed of lines and arches.

The key feature of our approach is that we automatically choose rule parameters according to physical constraints. In order to evaluate physical constraints some information from momentless shell theory will be needed.

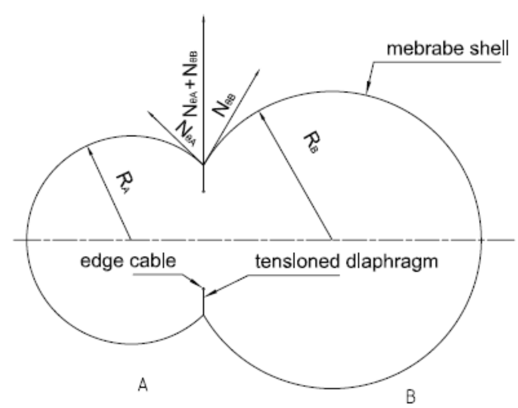

Figure 4. Coalesced spherical membranes. Force equilibrium along intersection line. Resulting tension is always perpendicular to axis of symmetry.

E. Initial Equilibrium Finding

The method selects rule parameters by determining values that will make the model stand in an equilibrium configuration. For axisymmetrical membranes, Eq. (1) leads to [20]

$$\frac{n_\theta}{R_1} + \frac{n_\phi}{R_2} = p_R \qquad (2)$$

where n_θ and n_ϕ are meridional and circumferential membrane tension* and p_R is the pressure normal to the membrane surface. In case of uniform internal pressure $p_R = p$ and using $R_0 = R_2 \sin\theta$ we have:

$$n_{\theta p} = \frac{p}{R_2 \sin^2\theta} \int_{\chi=0}^{\theta} R_2 \sin\chi R_1 \cos\chi \, d\chi$$
$$= \frac{p}{R_2 \sin^2\theta} \int_{\chi=0}^{\theta} R_2 \sin\chi \frac{d}{d\chi}\left(R_2 \sin\chi\right) d\chi$$
$$= \frac{pR_2}{2}$$

Using Eq. (2) and having in mind $p_R = p$, finally we obtain for circumferential membrane tension:

$$n_{\phi p} = pR_2 \left[1 - \frac{R_2}{2R_1}\right]$$

For a membrane in equilibrium the membrane tension is always nonnegative $n_{\phi p} \geq 0$ therefore,

$$2R_1 \geq R_2 \qquad (3)$$

That clear geometrical condition may be easily implemented into production rules. Thus, the shape obtained by application of this constraint on the minimum meridional curvature is expressing the physical condition for a non-wrinkled membrane. The condition (3) means that discontinuity cannot be maintained only by membrane action. An intermediate element should be inserted in places where discontinuity occurs. To define the type of discontinuity we will use terms accepted in pressure vessel engineering: *knuckle* and *flair* as illustrated in Fig. 3. In knuckle points a compression ring must be inserted, in flair points a tension hoop or diaphragm perpendicular to the axis of revolution. Figure 4 shows two coalescent spheres and the forces occurring in the intersection line. The membrane can not maintain out-of-plane forces. Therefore they should be borne by an additional element. In the example it is an edge cable supported membrane situated in the plane of the forces. We will denote such an element as a "diaphragm". A simple hoop cable is a possible solution as well.

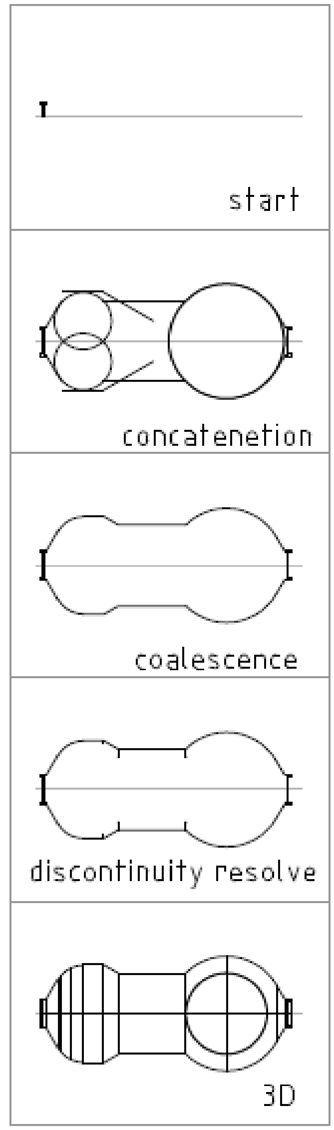

Figure 5. Shape grammar of pressure vessel.

F. Shape Grammar for Pressure Vessel Assemblage

Building models are most naturally constructed as a union of volumetric shapes[23].

1. 1D: Axisymmetric Case

Let us begin with a one-dimensional "string of shapes". Keeping axial symmetry we will be able to keep symmetry of stress distribution and to have a clear criterion on how to apply constraints of the membrane hypothesis. The shape grammar will be simplified to a grammar generating strings of symbols.

* In membrane theory, *membrane tension* is measured as force per unit length e.g. N/m. Stresses may be obtained by dividing membrane tension by thickness.

2. Generic Shapes

The simplest construction uses a sphere as the basic primitive. The first extension will include a cylinder, cone and torus segment presented by their generator curves: lines and arc respectively. Further surfaces of revolution are generated by conic section curves – paraboloid, hyperboloid, ellipsoid and isotensoid. The initial shape (axiom) must include the axis of revolution which is common for all shapes. The inherent coordinate system will have its origin at the center of *I* and the *x* coordinate will be the said axis of revolution.

Then a string of shapes (1D assemblage) is generated by semantically driven rules, arranged in a way as to satisfy typological requirements. As usual, volumetric primitives intersect each other. Intersection lines will be circles concentric to the axis of revolution and their longitudinal section will be presented as points. Due to axial symmetry the axial section is representative enough. Primitives will be presented by lines, circles and arcs, as far as primitives are limited to first extension. Their joining will represent the resulting shape. The next step after joining two entities is the discontinuity check. Symbolically it may be notated as a given function

discont($S_{i|}$, S_{i+1}) ::= { knuckle | flair | Ø }

and the respective procedural rule, in_case_of **discont**($S_{i|}$, S_{i+1})

; insert diaphragm
flair : (S_i, S_{i+1}) => (S_i, **Dp**, S_{i+1})

; insert compression ring (4)
knuckle : (S_i, S_{i+1}) => (S_i, **Cn**, S_{i+1})

; no change
Ø : (S_i, S_{i+1}) => (S_i, S_{i+1})

The shape dimensions can be either fixed or *parametric*. A parametric shape is commonly called a *schema*. In order to locate a specifically sized schema in a drawing, each schema has a transformation function associated with it. A shape grammar generates a design through the application of schema rules. A rule is in the form $S_i \rightarrow S_{i+1}$ where S_i and S_{i+1} are schemata.

To make the statement clear and the output more consistent, the geometrical parameters can be set within a limited range of discrete values – an approach applied in many parametric CAD systems.

The other constraint to be implemented in rules is the constraint on minimum meridional radius of curvature, as represented by inequality (3). Coaxial sphere, cylinder and cone always satisfy inequality (3). For the torus sector the condition is context-sensitive: it depends on the point where it is joined to the neighbor section.

The interaction between two generic shapes will be described first. Stresses and deformations introduced by internal pressure (as govern external load) can be determined for each part separately.

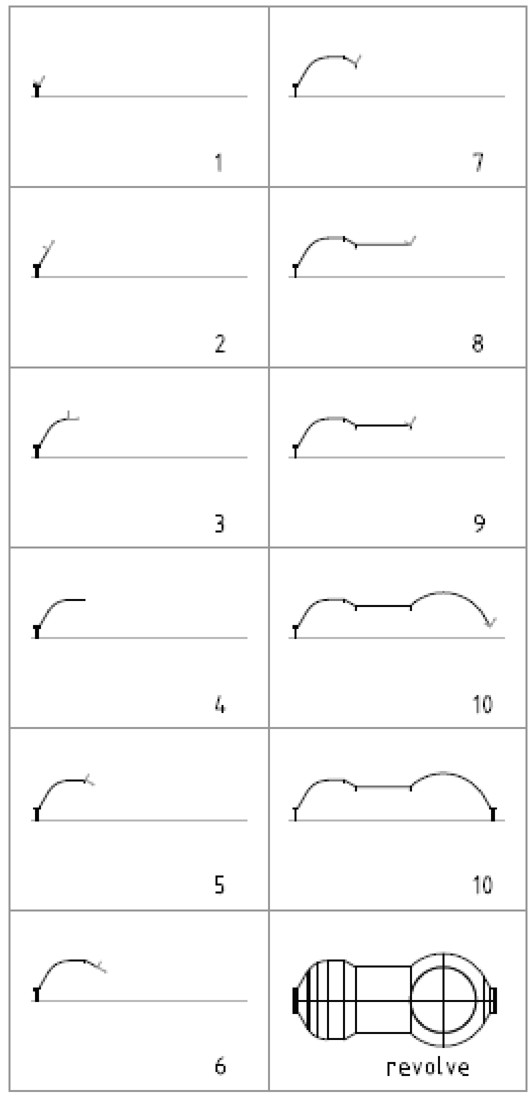

Figure 6. Parametric modeling of shell of revolution by its meridional curve.

As a first step, we propose a straightforward algorithm that applies physical constraints as conditional rules affecting directly generated shapes instead of a parametric search or optimization loop. Figures 5 and 6 illustrate the foregoing.

3. Alternative Implementation by Meridional Curve

Since the shell of revolution is entirely determined by its axis of revolution and its meridian it is possible to construct a 2D shape grammar with very simple rules generating the desired curve. The generator curve grows from initial point to terminal point by successive steps (sectors). Obviously, the meridian must have its starting and ending point laying on the axis of revolution to form a closed shell. The nonterminal symbol of such a grammar may be a local coordinate system associated with the end of the last (opened) section of the generator curve. Traditionally in shape grammars this has been done by using labeled points and lines to mark where the rule can apply. The discontinuity check will be directly obtained from parameters of the next section, specifically its starting angle with respect to local coordinates. In this way the rules will not be context-sensitive. Moreover, thus implemented grammar will be a *regular grammar*. Regular grammar has rules:

$$A \to u$$

where

$$A \in V \;\; ; \;\; u \in \Sigma$$

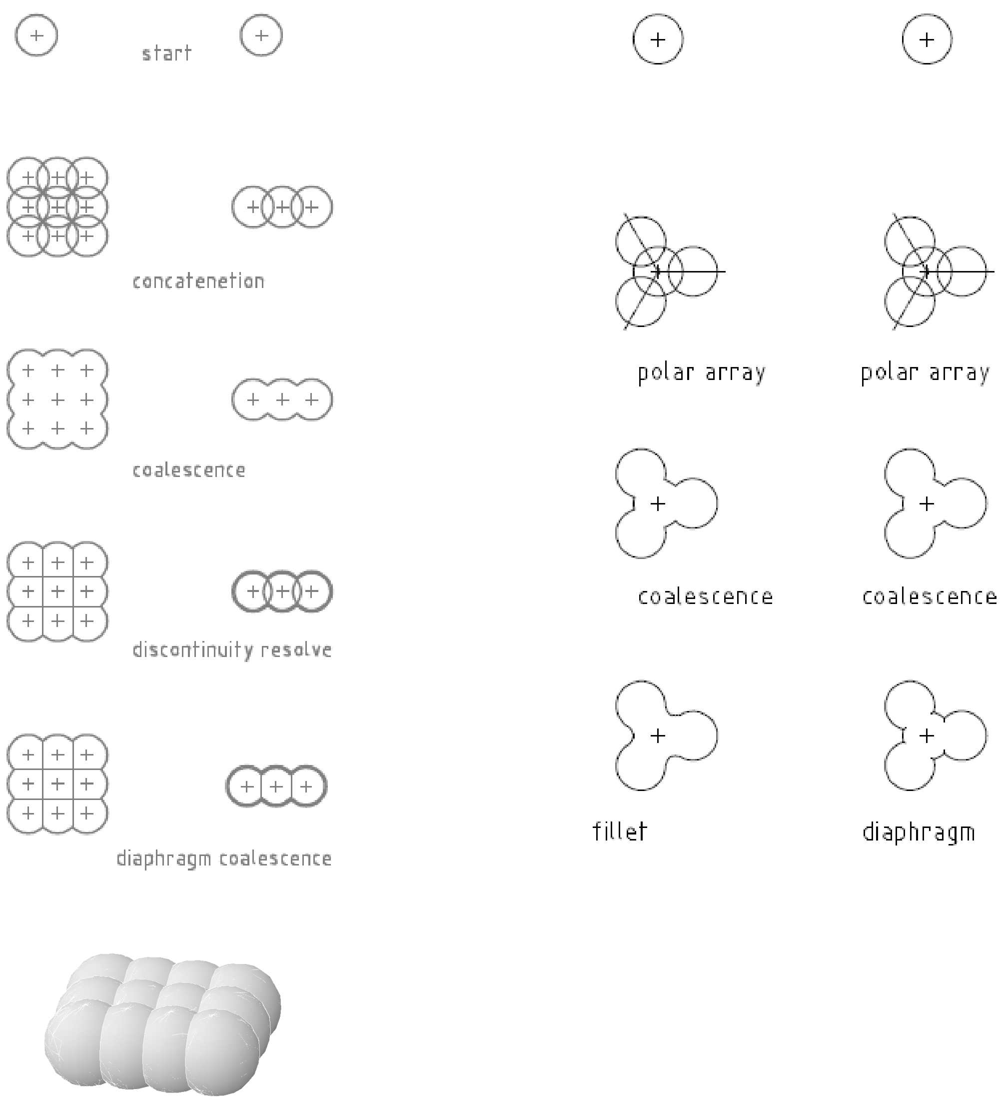

Figure 7. Rectangular and polar array of coalesced spheres.

In our framework they will be modified to parametric rules. The nonterminal symbol *A* is presented by the local coordinate system associated with the open end of the generated curve. Parameters should include the length of the next section and the radius of arc. A special rule will deal with discontinuities by means of rotation of local coordinates and adding an intermediate element – a compression or tension ring. The physical constraints will be implemented as constrained input parameters.

The growth of a generator curve and an example shape are illustrated on Fig. 6.

4. 2 and 3 Dimensional Generative Grammars

The next step is the extension of generative procedures in more dimensions. We will keep placing the objects along axes, using more then one axis. The most common axis topologies on a plane are: grid (array); rotational symmetry (polar array, star topology); and tree topology. An array of spheres constructs a well-known “mattress” as shown in Fig.7. The diaphragms as already shown in a row of spheres in Fig. 4 might be coalescent. That is not only a metaphor. Such an outcome can be obtained in the case of 3 coalescent bubbles (spheres) and placing diaphragms according to rules (4) the diaphragms will intersect each other. A procedure for joining the diaphragms, and for compensating open stresses, results in the formation of a foam-like structure. Thus openings and a net of diaphragms and tendons will appear instead of walls between the bubbles. The geometry of this internal structure must be designed by means of a finite element form-finding method such as Dynamic Relaxation or Force Density [24].

Transfer to 3-dimensional nesting will increase the complexity of interaction between shape primitives. A simple example of spheres nested on a space lattice is shown on Fig. 8. Using that technique a structure analogous to multiple-dome structures may be constructed. Whereas masonry domes and vaults are structures designed to be all in compression, the Macrofoam will be all in tension. An initial try toward implementation of that idea is illustrated in Figs. 8 and Fig. 10.

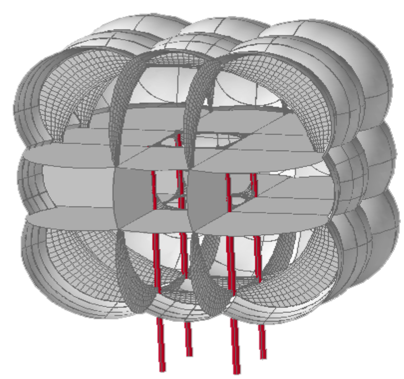

Figure 8. Internal structure of cubic lattice of coalesced spheres. Highly tensioned diaphragms and edge cables form secondary structure and provide shear stiffness of entire structure.

IV. Evaluation

Traditionally, standard 3D CAD packages, like Autodesk AutoCAD, Rhino 3D etc. offer many different tools to manipulate geometry, and therefore almost any structural system can be created within such program environment. The main disadvantage is the lack of physical constraints on geometry created by such tools. On the other hand, existing structural analysis tools focus heavily on providing analysis of the stress state of *already known* geometry. The “third way” is the solution of inverse problem – to find geometry when stress distribution is known is solved in the so-called *formfinding* process.

A principal step in membrane structure design is the finding of an initial equilibrium shape or the so called *form-finding*. Frei Otto noted that “only a fraction of all imaginable shapes can be formed pneumatically.” [22] Form-finding is a heuristic process of imaging pneumatically formable shapes based on strict physical constraints. Analytical models based on exact equation solutions are possible only for a small number of highly symmetrical shapes. A finite element analysis (FEA) computer model is needed for more complex shapes. The surfaces are divided into a number of small finite elements such as triangles, for example. Therefore, all possible geometries can be calculated. There are two theories: the linear Force Density Approach with links as finite elements and the non-linear Dynamic Relaxation Method with finite triangles [24]. Form-finding software for pneumatic structures is available as part of commercial packs for membrane structure design. Form-finding methods implemented in that software have an inherent limitation in control over outcome shapes [25, 26].

An alternative approach is to use a shape grammar with production rules that iteratively evolve a design by creating more and more details. The parametric nature of shape grammars, and their ability to deal with physical form rather than abstract elements, give them significant advantages over traditional production systems for geometry-based engineering design.

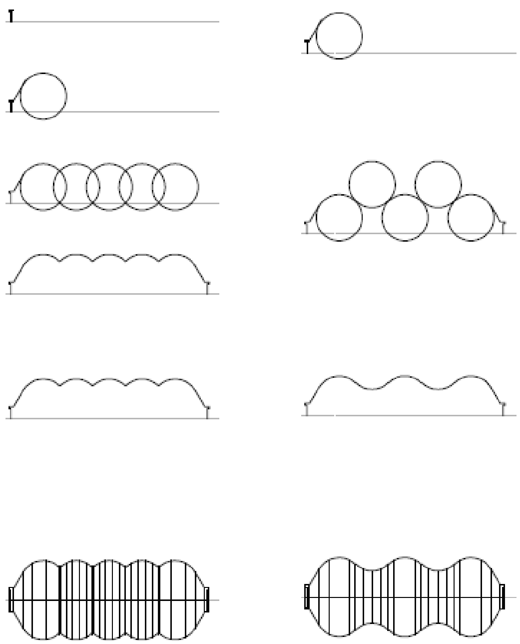

Figure 9. Lobed columns with minimum allowable radius of curvature R_1.

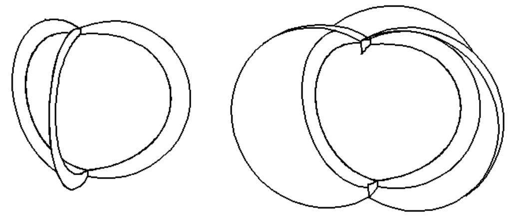

Figure 10. Three coalesced spheres and minimal diaphragms supporting discontinuity along intersection lines.

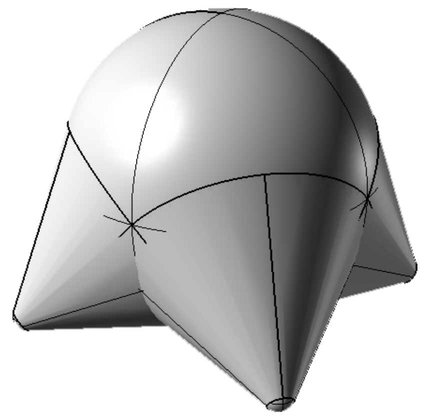

Figure 11. A multi-axis assemblage of shapes having lower symmetry than a sphere must include intermediate elements with a symmetry corresponding to symmetric spatial direction of the axes. Direct connection may disturb the equilibrium state.

Procedural generative systems such as Müller and Wonka [8] apply a split of the current shape into multiple shapes, repeating one shape multiple times, and component split creating new shapes on components (e.g. faces or edges) of the current shape. They are focused on detailing volumetric shapes. Our goal is to generate shapes rather than detail them. What may be misinterpreted as detailing: the intermediate elements introduced forward are strongly coupled with physical constraints and are a part of the primary structure rather than detail. Detailing, by adding different types of apertures – illuminators, airlocks etc. – is beyond the scope of the present text.

V. Discussion

Procedural modeling of space-based pressurized (inflatable or rigid) modules is one possible application of the described implementation of shape grammars. The other is the examination of the “syntax” of shapes intended to be applied as inflatable parts of space habitats and analysis of design rules of existing structures. For instance, the pressurized volume of the International Space Station may be parsed as a branch of orthogonal axes and rotational bodies generated along the said axes.

Rules in architectural literature are very powerful, but typically abstract and under-specified, so they can be applied only by humans. Shape grammars are a step toward more strict definition of particular rules. Software implementation needs more rigorously to define rules but that will result in reduced flexibility. Production systems as presented above are simple enough to be compiled by hand.

Traditionally, shape grammars treat shapes as non-atomic elements that can be decomposed and reassembled in an arbitrary way. This allows the application of rules to consequential shapes. This shape emergence is hard to be formalized and implemented in code. Even with just one rule for each shape the application of rules is undetermined because they can be applied to multiple shapes within a figure. The problem with computations using the specific algebraic representations of shape grammars is that they can be subject to ambiguity, combinatorial explosion and infinite numbers of emerging possibilities. This can be avoided by means of a set-based representation which doesn't support emergent features, making all rules decidable. In the framework of this study *emerging* is not explicitly used in favor of the possibility of computer implementation.

VI. Conclusion

This work proposes the use of shape grammars as the framework for a pressure vessel architectural design system. As first step the parameters that modify the output structure according to physical constraints are implemented in rules and no computation or user input parameters are needed. In further development standard form-finding procedures may be implemented in a parametric optimization loop.

Buildings, regardless of whether they are terrestrial or celestial, are most naturally constructed as a union of volumetric shapes. Creating and arranging in space are only a part of the process of combining generic pneumatic shapes to new pneumatic shapes. Because membrane structures are forming active structures, a number of solutions must be applied to redistribute stresses toward a new equilibrium state. Such solutions can be obtained through parametric generative rules. The user selects a set of free parameters filtered to reach a stable structure. Typical examples may be the radius of a next-step meridional curve or the width of a next primitive.

References

[1] Stiny, G. and Gips, J.. Shape grammars and the generative specification painting and sculpture. Information Processing, 71:1460–1465, 1972.

[2] Stiny, G., 1975. Pictorial and Formal Aspects of Shape and ShapeGrammars. Birkhauser Verlag, Basel.

[3] Stiny, G. and Mitchell, W. J., 1978. The palladian grammar. Environment and Planning B 5(1) 5 – 18.

[4] Koning, H. and Eizenberg, J., 1981. The language of the prairie: Frank Lloyd wrights prairie houses. Environment and Planning B 8, pp. 295–323.

[5] Flemming, U., 1987. More than the sum of its parts: the grammar of Queen Anne houses. Environment and Planning B 14, pp. 323– 350.

[6] Duarte, J., 2002. Malagueira Grammar – towards a tool for customizing Alvaro Siza's mass houses at Malagueira. PhD thesis, MIT School of Architecture and Planning.

[7] Stiny, G. 1980. Introduction to shape and shape grammars. Environment and Planning B 7, 343–361.

[8] Müller, P., Wonka, P., Haegle, S., Ulmer, A., and Gool, L. V. 2006. Procedural modeling of buildings. ACM Transactions on Graphics 25, 3, 614–623.

[9] Whiting, E., J. Ochsendorf and F. Durand, "Procedural Modeling of Structurally-Sound Masonry Buildings" *ACM Transactions on Graphics* Vol. 28, 5 (2009) p. 112.

[10] Ordway, F.I. III, Spaceships of the Imagination (in Collier's and 2001), in 2001: Building for Space Travel, Zukowsky, Ed., The Art Institute of Chicago, 2001.

[11] Vogler, A., Modular Inflatable Space Habitats First European Workshop on Inflatable Space Structures 21-22 May 2002 ESTEC, Noordwijk, The Netherlands http://www.spacearchitect.org/pubs/ESA-WPP-200.pdf

[12] Yin, P. K., A Preliminary Design of Interior Structure and Foundation of an Inflatable Lunar Habitat, Final Report, NASA/ASEE Summer Faculty Fellowship Program 1989, Johnson Space Center.

[13] Nowak, P. S., Criswell, M. E., and Sadeh, W. Z. (1990). Inflatable structures for a lunar base. Engineering, construction, and operations in space II, S. W. Johnson and J. P. Wetzel, eds., ASCE, New York.

[14] Dessauce, M., *The inflatable moment : pneumatics and protest in '68*, Princeton Architectural Press, New York 1999.

[15] Balbuch, L.I., N. A. Alfutov, V. I. Usyukin, *Stroitelnaya mehanika raket,* Visshaya Shkola, Moskva, 1984 (in Russian).

[16] (NASA-TM-X-733C5) Astronautic Structural Manual, Vol. 2, NASA (section 7, Thin Shells).

[17] Moss D. R.*, Pressure vessel design manual: illustrated procedures for solving major pressure vessel design problems*, Amsterdam, Boston, Elsevier/Gulf Professional Pub., 2004.

[18] Bednar, H.M. *Pressure Vessel Design Handbook*, Van Nostrand Reinhold, New York 1986.

[19] Jong M. de, A. Lennon, Pressure Restraint Design for Inflatable Space Habitats, 10th European Conference on Spacecraft Structures, Materials and Mechanical Testing, 1st CEAS European Air and Space Conference, Berlin 2007.

[20] Flugge, W. *Statik und Dynamik der Schalen*, Springer Verlag, Stuttgart, 1957.

[21] Timoshenko, S. and Woinowsky-Krieger S. *Theory of Plates and Shells*, McGraw-Hill, NY, 1959.

[22] Otto, F. (Hrsg.) (1962) *Zugbeanspruchte Konstruktionen. Gestalt, Struktur und Berechnung von Bauten aus Seilen, Netzen und Membranen. Band I Pneumatische Konstruktionen*, Ullstein, Frankfurt, Berlin

[23] Le Corbusier, Towards a New Architecture. Dover Publications, 1985

[24] Gründig L., et al. (2000), A History of Principal Developments and Applications of the Force Density Method In Germany 1070 1999 ,IASS-IACM 2000, Fourth International Colloquium on computation of Shell & Spatial Structures, June 5-7, 2000, Chania-Crete, Greece

[25] Kröplin K., R. Wagner, (2004). On the Applications, the Potential and the Limits of Inflatable Membrane Structures, ECCOMAS 2004 Jyväskylä, 24—28 July 2004

[26] Stavrev V. Creative formfinding for inflatable and pressure stabilized structures for space habitats, proceedings of ECCOMAS Structural Membranes, Stuttgart, 2009

Appendix

Additional literature on implementation of a shape grammars:

Tapia, M. (1999). A visual implementation of a shape grammar system. *Environment and Planning B: Planning and Design 26*, 59-73.

Stiny, G. 1982. Spatial relations and grammars. Environment and Planning B 9, 313–314.

Piazzalunga, U. and Fitzhorn, P. Note on a three-dimensional shape grammar interpreter. Environment and Planning B: Planning and Design, 25(1):11–30,1998.

Chau, H.H., Chen, X., McKay, A., & Pennington, A. (2004). Evaluation of a 3D shape grammar implementation. *Proc. 1st Int. DCC Conf.*, pp. 357- 376. Dordrecht: Kluwer Academic.