

NUMERICAL MODELS OF SPECIAL TYPES OF SPATIAL STRUCTURES DEFINED IN FORMIAN

Janusz REBIELAK*

* 1. Chairman of the Committee for Architecture and Town Planning
Wroclaw Branch of the Polish Academy of Sciences, Poland.

2. Retired Professor of Faculty of Architecture,
The Tadeusz Kosciuszko Cracow University of Technology, Poland

E-mail: j.rebielak@wp.pl URL: <http://januszrebielak.pl>

Keywords: *architecture, numerical model, programming language, spatial structure, aircraft structure*

ABSTRACT

Among the many definitions of spatial structures, the simplest is the one that defines such a structural system as consisting of many components distributed relatively evenly in its space, and the transfer of forces between components also takes place in a spatial manner. Spatial structures are most often made with the use of straight members forming many types of spatial trusses, which are multifold statically indeterminate systems [1-3]. The use of numerical models defined in the Formian programming language greatly improves and accelerates the process of designing these complex types of the engineering structures [4], which moreover are also characterized by the self-contained architectural form.

BASIC PRINCIPLES OF FORMEX ALGEBRA AND FORMIAN

The programming language Formian has been invented by Prof. Hoshyar Nooshin on the basis of principles of “formex algebra” and then it was gradually developed by himself and his research teams. It enables a parametric way of defining numerical models of even very complex form of structural systems. In “formex algebra” the three dimensional space has a co-ordinate system of three axes the U1, U2 and U3, what is similar and congruent with the typical co-ordinate system of the three axes X, Y and Z. Systems of “normat points” and “normat lines” are located on the flat surface U1U2, see Fig. 1a. The same remarks refers to “normats” placed on surface U1U3 and on surface U2U3. Distances between particular “normat lines” are individually determined by the author of the Formian’s program. The composition of for instance the two

straight truss members AB and BC, presented in Fig. 1a, is a “formex variable” ABC defined in Formian as follows: $ABC=\{[1,0;0,1],[0,1;1,2]\}$;

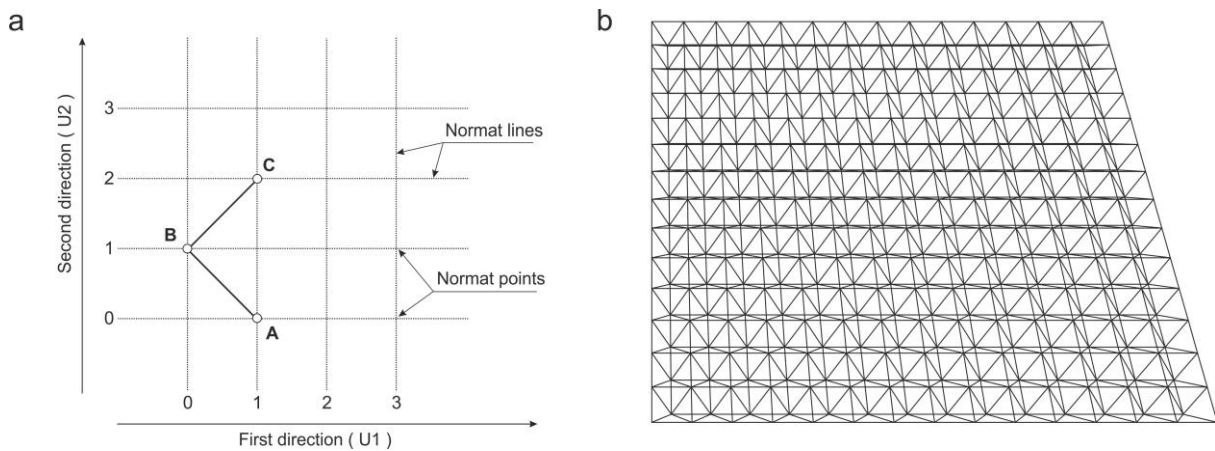


Fig. 1. a) Example of reference system applied in “formex algebra” for algebraic definition of geometrical composition in space of a structure together with its “formex”, b) visualization of a flat form of a double-layer space frame „square-on-square” determined by a numerical model defined in the programming language Formian

In the Formian programming language numerous types and groups of functions are used, and even the application of a few simple functions belonging to the “Cardinal functions” enables the definition of numerical models of an example form of a spatial truss, which is the double-layer “square on square” space frame, see Fig. 1b.

NUMERICAL MODELS OF COMPLEX BUILDING STRUCTURES

Although the flat form of a spatial structure, shown in Fig. 1b, is its simple geometric form, but the structure itself is a complex structural system. Numerical models of more complex geometric forms of spatial structures are defined using specialized functions of the Formian language. Defining numerical models of geodetic forms require the use of such functions. The first one is the “polymation” function, which is used to determine the positions of the structure components on the flat faces of the polyhedron, adopted as the basic one in the process of shaping of the geodetic structure. The second one is the “tractation” function, whose task is to properly project the forms defined on the flat faces of the basic polyhedron onto the concentric sphere. These two functions are applied to define numerical models for single and multi-layer geodesic structures. In a Formian program, the “polymation” function is abbreviated as "pol" and the “tractation” function has the abbreviation "trac". These two functions were used in Program A, shown below, defining the numerical model of the main structure of the Centrum Geo, see Fig. 2.

Program A

```

01. m=6; (*)Number of subdivisions of icosahedron edges(*)
02. s=1/3*sqrt(3);
03. U1=1.63299316;(*)3rd coordinate for the upper layer(*)
04. M2=0.0;(*)3rd coordinate for the middle layer(*)
05. L3=-1.63299316; (*)3rd coordinate for the lower layer(*)
06. T1=rosad(2,2*s,3,-120)|[2,0,M2,2;3,3*s,M2,2];
07. TP1=genic(1,2,m/2,m/2,4,6*s,2,-1)|T1;
08. T2=rosad(4,4*s,3,-120)|[3,3*s,M2,2;5,3*s,M2,2];
09. TP2=genic(1,2,m/2-1,m/2-1,4,6*s,2,-1)|T2;
10. TG1=rosad(2,2*s,3,-120)|[2,0,M2,2;2,2*s,U1,1];
11. TGW1=genic(1,2,m/2,m/2,4,6*s,2,-1)|TG1;
12. TG1=rosad(2,2*s,3,-120)|[2,0,M2,2;2,2*s,U1,1];
13. TGW1=genic(1,2,m/2,m/2,4,6*s,2,-1)|TG1;
14. TG1b=rosad(2,2*s,3,-120)|[0,0,M2,2;2,2*s,U1,1];
15. TbGW1=genic(1,2,m/2,m/2,4,6*s,2,-1)|TG1b;
16. N1={0,0,M2,2},[2*m,0,M2,2],[m,3*m*s,M2,2];
17. TbGWL=lux(N1)|TbGW1;
18. TD2=rosad(4,4*s,3,-120)|[3,3*s,M2,2;4,4*s,L3,3];
19. TDP2=genic(1,2,m/2-1,m/2-1,4,6*s,2,-1)|TD2;
20. TD2b=rosad(4,4*s,3,-120)|[4,0,M2,2;4,4*s,L3,3];
21. TDP2b=genic(1,2,m/2-1,m/2-1,4,6*s,2,-1)|TD2b;
22. TRnr2A=TP1#TP2#TGW1#TDP2#TbGWL#TDP2b;
23. EdPs=rosad(m,m*s,3,120)|{[2,0,M2,2;0,0,M2,2],[1,3*s,M2,2;0,0,M2,2],
[2,2*s,U1,1;0,0,M2,2],[0,0,M2,2;0,0,M2,2]};
24. Seg2s=TRnr2A#EdPs;
25. PA=rin(1,m-2,2)|[2,0,M2,2;4,0,M2,2];
26. Edge=lam(1,6)|lam(1,4)|[3,3*s,M2,2;4,0,M2,2];
27. TTT=Seg2s#TRnr2A#PA# Edge;
28. ws3=0.55;(*)Parameter1(*)
29. FaceP1=bt(1,1,ws3)|TTT;
30. Rg=18;
31. Rm=Rg-U1*ws3;
32. RL=Rg-2*U1*ws3;
33. G1=pol(1,5,'[1-15]',[3,4;1,Rg;2,Rm;3,RL],[0,0;2*m,0],
[0,0;2*m,0],[0,0;2*m,0])|FaceP1;
34. Edge=rin(1,m,2)|[0,0,M2,2;2,0,M2,2];
35. G2=pol(2,5,'[21-25]',[3,4;1,Rg;2,Rm;3,RL],[0,0;2*m,0],
[0,0;2*m,0],[0,0;2*m,0])|Edge;
36. G1=pol(1,5,'[11-15]',[3,4;1,Rg;2,Rm;3,RL],[0,0;2*m,0],[0,0;2*m,0],
[0,0;2*m,0])|FaceP1;
37. TTG=Seg2s#TRnr2A#PA;
38. FaceP2=bt(1,1,ws3)|TTG;
39. G3=pol(1,5,'[1-10]',[3,4;1,Rg;2,Rm;3,RL],[0,0;2*m,0],[0,0;2*m,0],
[0,0;2*m,0])|FaceP2;
40. G3=G1#G2#G3;
41. G15=trac([3,4,1,2,3],1,[0,0,0,Rg,0,0,0,Rm,0,0,0,RL],4,-15)|G3;
42. draw G15;(*) See Fig. 2 (*)

```

Program line numbering is only used in Formian-K, the latest version of this programming language. The main argument of Program A is given on its line 41, where the “tractation” function is used. The “polymation” function has been used in several lines with the numbers 33, 35, 36 and 39, respectively. These functions have a seemingly simple notations, but their mathematical operation is very complex and requires considerable knowledge and experience from the programmer. These two functions were also applied in the process of defining a numerical model of a complex of tall buildings called GeoDome Sky Towers, in the central part of which a spherical structure with a diameter of 100 meters was designed, Fig. 3. This guarantees the definition of numerical models of very complex forms of construction systems, which models are the basis for the process of their design and implementation of the appropriate visualizations.

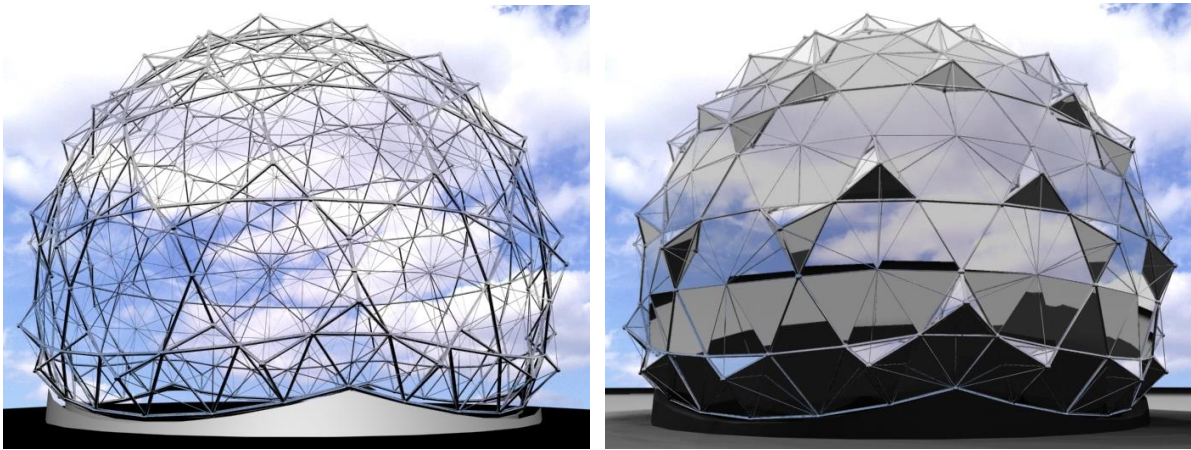


Fig. 2. Visualizations of the numerical model of the VA(TH)No2 tension-strut structure defined in the programming language Formian for the Centrum Geo, (Anna Drozd)

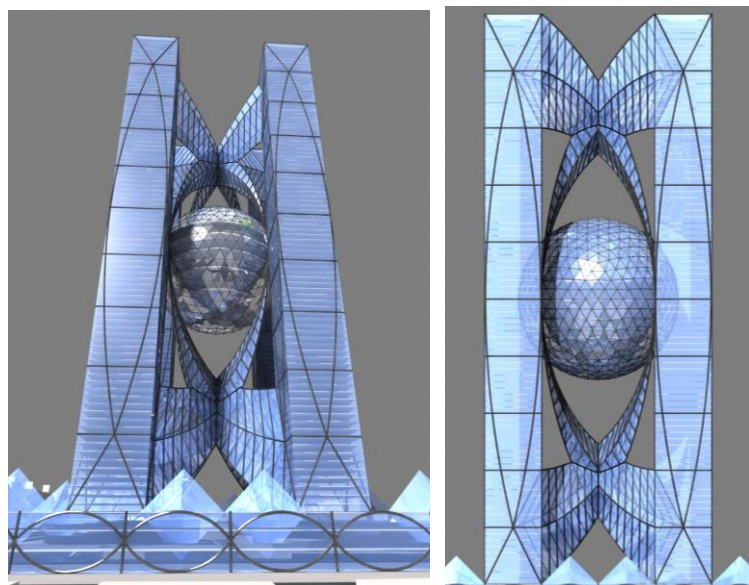


Fig. 3. Visualizations of the numerical model of the GeoDome Sky Towers defined in the programming language Formian, (Maciej Smoliński)

NUMERICAL MODELS OF COMPLEX AIRCRAFT STRUCTURES

Aircraft structures must, above all, be light when subjected to very high values of forces caused mainly by the dynamic loads. The structural efficiency of such structures was in the 1960s the inspiration for Fazlur Khan [5] to develop very light and efficient construction systems for the very tall buildings, which are also successfully used today. The author has been an aviation enthusiast since his youth and is involved in the development of concepts, among others, of the vertical take-off and landing aircraft, of course using their numerical models defined in Formian. Visualizations of numerical models of two types of such airplanes are presented in Fig. 4 and in Fig. 5 [6,7].

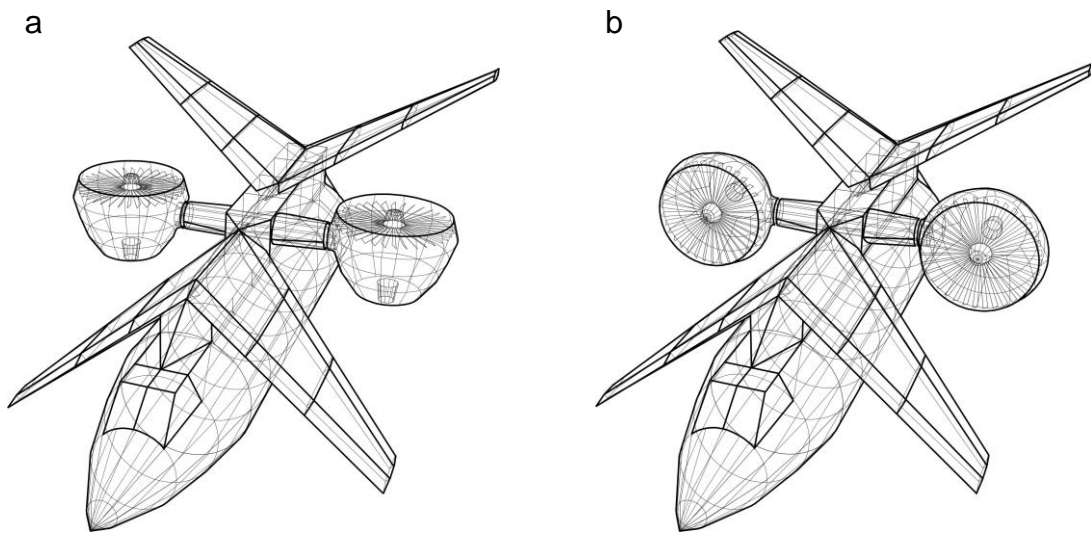


Fig. 4. Visualizations of the numerical model of the mono-fuselage airplane structure with a dorsal stringer

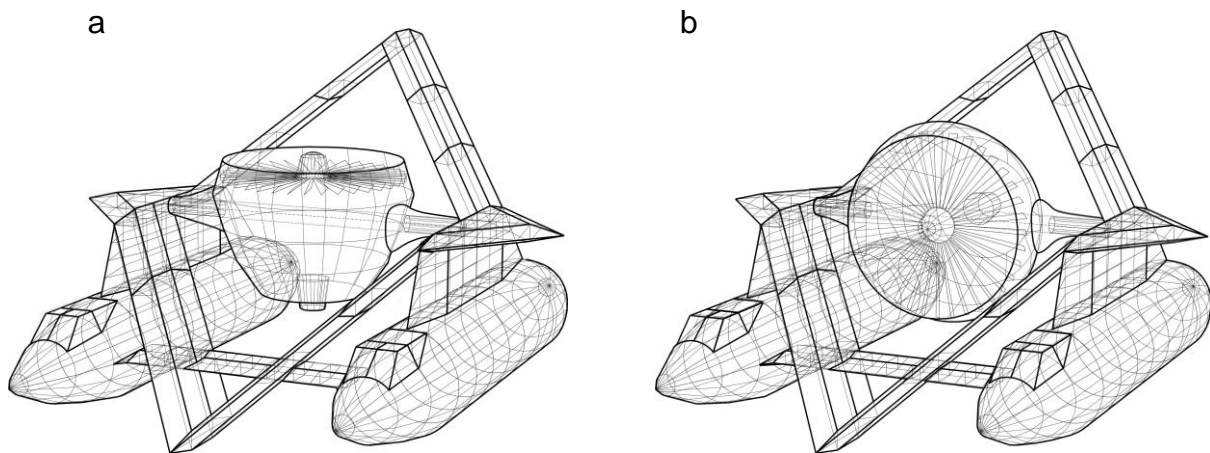


Fig. 5. Visualizations of the numerical model of the double-fuselage airplane structure with dorsal stringers

The design of the vertical take-off and landing airplane with a dorsal stringer according to the invention [6] enables the processes of safe and precise vertical take-off and landing, as well as performing horizontal task flights at various altitudes and in a relatively large range of cruising speeds. During take-off and landing, the axes of the engines are vertically oriented, Fig. 4a, Fig. 5a, and after reaching the appropriate height and the appropriate rotation of the engines by 90 degrees, the plane continues the flight in a horizontal configuration, Fig. 4b, Fig. 5b. With an indirect inclination of the engines of 45 degrees from the horizontal, this aircraft also obtains the characteristics of a short take-off and landing aircraft. The controlled process of the vertical take-off and landing of the aircraft is mainly ensured by the use of appropriate designs of the engine exhaust nozzles enabling precise and immediate thrust vector control of each of its jet engines.

CLOSING REMARKS

Due to simplicity and other mathematic features of “formex algebra” the numerical models defined in the programming language Formian are very easy to modify in order to adjust the structural shape to meet directives at every design stage. The data generated by the application of Formian may easily be submitted as an input to chosen analysis packages. The application of Formian can enormously accelerate the design process, makes easier the cooperation between all participants of it and enables selection of the best, optimum structural configuration among numerous possible technical solutions.

REFERENCES

- [1] H. Nooshin, P. Disney and C. Yamamoto: *Formian*. Multi-Science Publishing Co. Ltd., 1993.
- [2] H. Nooshin, P.L. Disney and O.C. Champion: Computer-Aided Processing of Polyhedric Configuration, *Beyond the Cube. The Architecture of Space Frames and Polyhedra*. Gabriel J. F (ed.), John Wiley & Sons, 1997, 343-384.
- [3] H. Nooshin, P.L. Disney: Formex configuration processing II, *International Journal of Space Structures*, 2002; **17**; 1-50.
- [4] J. Rębielak: *Shaping of space structures. Examples of applications of Formian in design of tension-strut systems*, Oficyna Wydawnicza Politechniki Wrocławskiej, Wrocław, 2005.
- [5] https://en.wikipedia.org/wiki/Fazlur_Rahman_Khan.
- [6] J. Rębielak: *Vertical take-off and landing aircraft*, Patent Application No. P.438888, Patent Office of the Republic of Poland, 2021.
- [7] J. Rębielak: *Numerical model of structure of the vertical take-off and landing aircraft*, Conference materials of the XXV International School of the Computer Aided Design, Manufacturing and Maintenance, Jurata, September 13-17, 2021, Military University of Technology, Warsaw, 2021.