Parametric Description of Bridge Structures

Casimir Katz
Civil Engineer
SOFiSTiK AG
Oberschleissheim, Germany
casimir.katz@sofistik.de

Casimir Katz, born 1954, received his civil engineering degree 1977 and his doctoral degree 1982 from the Technical University of Munich. He lectures civil engineering informatics at the Technical University of Munich.

He is one of the founders and now CTO of SOFiSTiK AG.

Summary

As information is inevitably lost by computing, we often have the problem that during our design process relevant information is missing. One possibility to overcome this is to use parametric descriptions. To apply this technology, there are several techniques available. To design bridges, classical approaches are insufficient, and more sophisticated approaches are necessary.

Keywords: Parametric Descriptions, Bridges, Cross Sections.

1. Introduction

It is a fundamental law of any arithmetic operation, that information is lost in the process of computing. After adding two numbers like 71+84 = 155, no one will be able to reconstruct the two numbers just from the result 155. Cryptographic systems are based on the principle that it is extremely difficult to find two prime factors of a very large product, if the factors are not prime, no unique solution exists. So whenever a pocket calculator is used to enter the result of an operation into the computer, information is lost at that same moment.

But there is also another effect when modelling a real system on a computer. It is necessary to do a lot of projections by stripping information which might be not relevant for the further processing. A structural model is deducted from the true world, and simplified by using a distinct theory. Then this model is discretised by a finite element mesh etc. This is similar to creating a shadow of a 3D body which contains less information than the original object.

Sometimes it does matter that information has been lost. If the structural system has become a polygon approximation to a curved true shape, it will become difficult to calculate centrifugal forces or the effects of post stressing tendons. Thus, it might be advantageous to retain as much information as possible for the later processing steps. This can be supported by parametric modelling. The main benefits for this type of modelling for bridges is not the freedom of the architect to create any structure, but the possibility for the engineer to optimise his structure and to have consistent data until the very end of his task.

2. Parametric Macro Languages

In the early days of computing, the input had to be provided with punched cards containing every single datum explicit and in a fixed form. All generation facilities had to be programmed explicitly in the program reading the input. With the introduction of interactive terminals, it became necessary to switch to a free format, and software developers extended the newly created format by simple generation facilities [1]. The next step was then to allow some programming logic, arithmetic expressions and variables [2]. Experienced users have a great preference for this input because it allows them to reuse their input data for similar problems, and it allows them to keep the information in the input. Quite often, those input data sets have been developed over years to become an interactive macro and contain a lot of company knowledge.

2.1 Basic Elements of a Powerful Macro Language

The basic elements are that input data may be described in a free format allowing arithmetic expressions and comments. Comparing the two statements:

```
SECT   1 0.24 0.0072
```

and

```
SECT   1 A    0.4*0.6    IY 0.4*0.6**3/12  $ Rectangular section 0.4/0.6
```

The difference is obvious: The second line still contains all relevant information. But the data is given three times, so the change of the dimensions is difficult and error prone, especially if the same datum is used at several locations in the input data.

So the next step should be to introduce a variable like:

```
STO#B 0.4    $ Width of right girder
STO#H 0.6    $ Height of right girder
SECT   1 A    #B*#H    IY #B*#H**3/12 $ Rectangular section girder 1
```

It is of course not only a matter of taste but also of computing knowledge to define a powerful syntax. Although many people prefer to use variables without any specific identifier, it might be advantageous to have a key character like the "#" in the example above to allow other programming keywords to be used without interfering with the variable name space. All examples in this context are given in the syntax provided with the CADINP-Language of SOFiSTiK, which is based on an assessment of commonly used syntax in the 1970s and 80s [1,2].

2.2 Extended Elements of a Powerful Macro Language

Without going into the details we need some more elements to allow the definition of complex tasks. The most important are control structures like an IF-ELSE-ELSEIF-ENDIF, loops with exit possibilities and of course the full set of algebraic functions like sine, cosine, square root etc. Even more functionality may be added by automatic unit conversions (e.g. imperial, metric, SI) by appending the correct unit at the item defined like "12.5[mm]". Three aspects however should be addressed in more detail:

It is an indispensable feature to have not only scalar variables but also arrays. Very important is not only the use within loops, but also the implicit information of the length of an array. With a little extra effort it is also possible to interpolate within an array or to combine two arrays within a table where interpolation is possible. If, for example, a width of a bridge is given at specific locations X along a bridge, a syntax like the following defines a function to interpolate the value:

```
STO#XTAB 0.0,2.0,10.,12.  $ x-positions along the bridge
STO#WTAB 6.0,4.0,4.0,5.5  $ width of top deck
STO#W   "TAB(XTAB,WTAB)"

STO#N 10
LOOP#1 #N+1 ; LET#X #1*#LENGTH/#N
   SECT 1+#1 A #W(#X)*#H   IY #W(#X)*#H**3/12 $ sections girder 1
ENDLOOP
```

The next step is to allow user defined functions. The interpolation specified here may be enhanced by quadratic or cubic interpolation using derivatives, but the most general form is to let the user specify the formula on his own:

```
STO#W   "FUN(X,MIN(6,3.0+0.123*(#X/#LENGTH)**2+COS(2*#PI*#X/#LENGTH))"
```

Finally, it is often advantageous to allow the use of data calculated by previous steps of the analysis. One example is the definition of loadings based on dynamic or buckling eigenvalues of the structure, but there are also cases where such a value depends on geometric shapes.

3. Parametric CAD Software

Parametric CAD software has been mainly used in the field of mechanical engineering. Parametric Technology Corporation PTC was one of the first software vendors to apply parametric modelling to engineering design, but in architectural and civil engineering has been quite uncommon. Today much more information about parametric modelling is available. The basics are very similar to the approach described before. We have named variables and mathematical formulas in dimensions using also named variables and arrays of components along a line or a surface and the exchange of components and properties.

Fig. 1 *Examples of buildings with parametric modelling (taken from Autodesks Revit Homepage).*

Nowadays, for buildings there are several software packages available providing powerful parametric solutions. For example the software package Revit from Autodesk has not only introduced a parametric building modeller with a parametric change engine, but also the concept of intelligent objects. For example, take a door object in a building. The data describing this door provides the information needed to show the use of the door in a 3D model as well as how it looks in either a plan view or elevation drawing. It also contains information that facilitates incorporating the door into a schedule that is added to a drawing sheet. The specific usage information describes the size of the door, its finish, the wall in which it is inserted, the distance from the end of the wall to the edge of the door and whether the location of the door is fixed or it moves if the end of the wall moves.

It is these latter characteristics that define a parametric design solution. If a designer changes the geometry defining the wall, the adjacent walls will expand or contract depending upon the change the user makes. Likewise, the door will move to a new position if its location has been tied to the top end of the side wall. Assuming the designer is working on a multi-storied building, changing the radius of a curved wall will affect the entire building, not just the floor on which the change is made.

The information captured by a parametric design system is often referred to as design intent. An example of this is the ability to lock a dimension in the design, specifying that an interior wall is always some distance from the opposite wall.

So the designers can change the geometry by either dragging an element such as a wall to a new location or they can change the value of the dimension and watch the model adjust to this new value. Because the dimension is an intelligent component, designers can place locks on various sets of dimensions to add design intent to their parametric building models. Design intent for setback requirements, code issues, or critical dimensions for specific spaces can be also managed.

The main difference to the approach described in the chapter before is that the dependencies are collected as a set of rules and that the software monitors all possible conflicts and warns the user if a change or definition violates a constraint.

Most civil engineers are overwhelmed at the first contact with all those features now available, believed to be impossible in the past.

However, it is a tremendous task to implement all those intelligent objects, and many bridges are unique, so it is an open question what can be done with this type of software.

4. How to Model a Bridge

The very fundamental question is now how best to model a bridge. There are very different types of bridges, of course. For a long suspension bridge, other approaches may be useful instead for a short integral bridge. But two fundamental approaches may be identified.

4.1 3D Solid Modelling

For integral bridges and architectural driven concepts it is most natural to develop a true 3D solid model of the bridge. This allows all freedom for the modelling of the bridge and it will be needed anyhow to create the drawings, the scaffolding and the final cost reports. It also nicely fits to the civil engineering tasks of the road planning and the digital surface model.

There are a lot of software packages designed to allow the exchange and viewing of product data, like the jt2go software (http://www.jtopen.com or http://www.plmxml.org) These packages allow the exchange of hierarchical definitions of a complex system and also the generation of sections across the given models, and the software provider UGS claims to go beyond parametric modelling: "System-based modelling utilizes a product definition template that captures the system level design parameters and intelligently establishes interfaces between various systems within a product". That sounds great, but we do not know if this matches our needs. So the first critical question is, if we can master the complexity of this approach. The second is if it will supply the features we really need.

However there is also one serious drawback. To allow the structural analysis of the bridge we have to either use a 3D solid finite element structure or we have to create a simplified structural system. Although some modern CAD software systems (Revit, Tekla) provide a facility to maintain such a structural system within the total CAD model, the generation of such a system is quite beyond an automatic generation in most cases. However there is software available, like COSAR which has a an integrated expert system SolidDesigner, claiming to analyse the 3D-geometry model and to generate a shell-midplane-model. (http://www.femcos.de)

Such a shell-midplane-model may be useful to model secondary effects like the transverse bending, local stress distributions, warping torsion, shape deformation or similar. However it is not easily generated, especially for those structures where different parts intersect in complex joints.

More often, a one dimensional reduction to beam elements is required. Although every section may be created easily by calculating the intersection of a plane perpendicular to the beam axis with the solid model, a lot of information needed for the analysis is not transferred to the beam model. So at least some inheritance of properties from the original model will be required.

4.2 Axis Bridge Modelling

The other concept which has been adopted not only by major software vendors in the field of bridge engineering but is also the base of the suggested IFC-Bridge definition is the reference of all data to an axis (e.g. IfcBridgeReferenceLine) given by classical elements of road design. The geometry is described separatly for the plan view (straight lines, circles and transition elements like clothoidal spirals, Bloss-Curves, sinusoidal or cosinusoidal curves) and the definition of the heights (straight and quadratic parabola) and the cross fall related to that same axis.

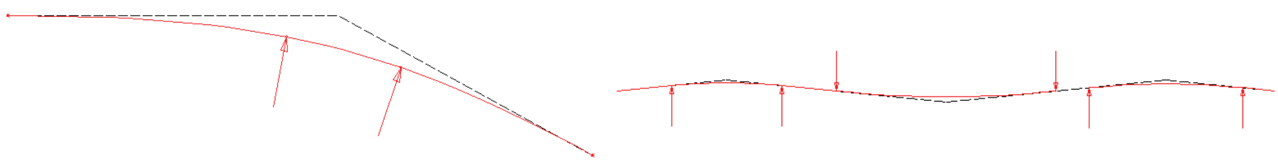


Fig. 2 *Bridge Axis – Plan view and elevation*

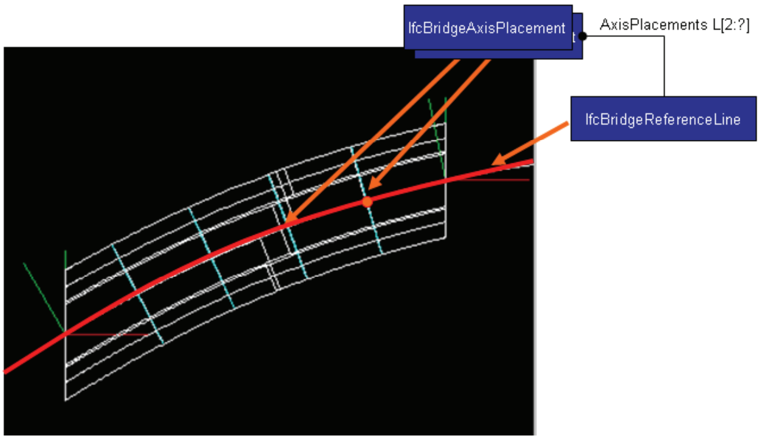


Fig. 3 *IFCBridge definitions*

All the specific entities can then be placed relative to the axis using a curvilinear coordinate s along this reference line. This reference line can also be the axis of the road on top of the bridge. S may be then taken as the distance along that central axis.

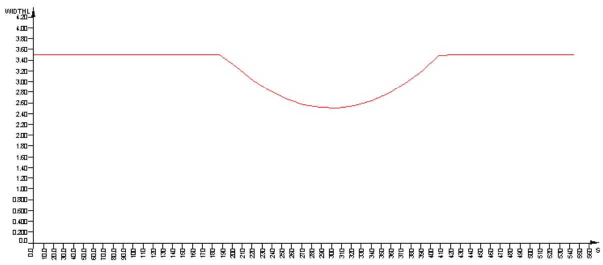
We must distinguish between the sections which are always taken perpendicular to the axis and general placements which may be inclined relative to the axis and rotated arbitrarily. There is a small problem with the construction of skewed sections, because we have to distinguish if the s value should be taken from the perpendicular point on the axis or from the location of the placement on the axis. The latter has advantages for points which extend before and beyond the range of the axis definition and we therefore recommend that approach.

Geometry relative to an axis may be defined with an offset. It is self evident that all the associated parameter curves (x(s) , y(s) , z(s)) should use the same metric of s as the original reference axis. So all the curves have a different true length and reference length on the reference axis. However, it should be noted, that a parallel curve to a clothoid is no longer a clothoid itself. Thus, we have to transform it to a general NURBS curve introducing some geometric errors, or perform all geometric calculations based on the reference geometry and an relative offset.

Finally, it is preferable to have an additional relative metric between the supports, defined as 0.0 at the start face, 1.0 at the first support, 2.0 at the second support etc. This allows easy definition of mid span locations or loaded ranges on the bridge.

4.3 Variables along the axis

So far the IFC-model does not contain any parametric definitions. Of course most of the geometric objects are included as references to the architectural model, but the essential ingredient for a parametric modelling which is missing in this draft is the definition of all possible parameters along the reference axis.



Again we might use linear, quadratic or cubic interpolation between given values and we might use formulas (see Fig. 4)

The use of the relative metric described in the chapter before allows the specification of the s-coordinate of a given value or even placement relative to a support by a special function (e.g. #SS(2)-#A where #SS is an reserved name for the table of the accumulated span widths)

Fig. 4 *Example of a parameter along an axis*

Again it is very useful to define all parameters as a function of other parameters, e.g. an offset to an already defined complicated parameter value.

We have now obtained the possibility for two sets of variables. If we allow a hierarchy in the name spaces we get a powerful feature allowing to define default values which have to be overwritten only for those parts of an axis where it is necessary. And we might transfer parametric descriptions easily between several reference axes or projects.

5. Definition of Sections with Parameters

It is straight forward to define absolute and relative dimensions of a section using formulas with parameters defined along the reference axis. However, there is more that can be done to define sections by parameters. To make the section more "intelligent", it is necessary to add some simple geometric operations, which are difficult to describe with formulas only.

5.1 Relative Coordinate Constructions

Many coordinates of the section are defined relative to other points. Typical examples are coordinates of reinforcements, shear cuts or stress locations. Some coordinates may be calculated with a directional mode, i.e. a reinforcement will always have a constant distance to another line, and the start and end point may be dependant on vertices of the section.

There are some possibilities to model those dependencies: one is to record the construction process (i.e. two offsets of lines, the intersection of which defines the coordinates of the desired point). The other is to define special parametric rules allowing the definition of the point. The latter may be easier and faster to implement and it allows much better control of nested dependencies, but there is clearly no unique solution to those problems.

Possible rules are:

- One or two coordinates are taken **relative** to one or two reference points, e.g. the y-coordinate may be taken from the first reference point, while the z-coordinate is taken from the same or another reference
- One or two coordinates are taken **absolute** from two reference points, e.g. the y-coordinate is taken from the first reference point, while the z-coordinate is taken from another reference
- The new location of the point follows a point and its direction, where none or a single or both coordinates are scaled according to the new position:

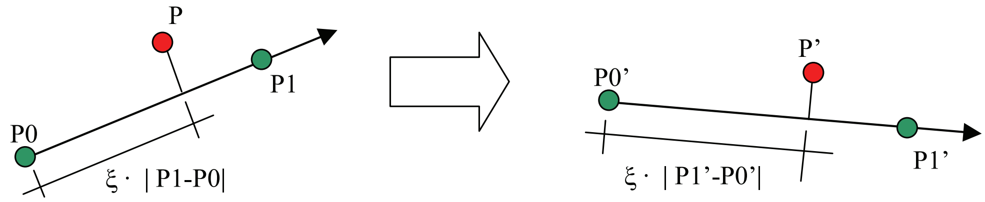


Fig. 5 *Directional references with two points*

- The new location of the point is calculated from a third point either as the perpendicular point or as the point with the same y or z coordinate:

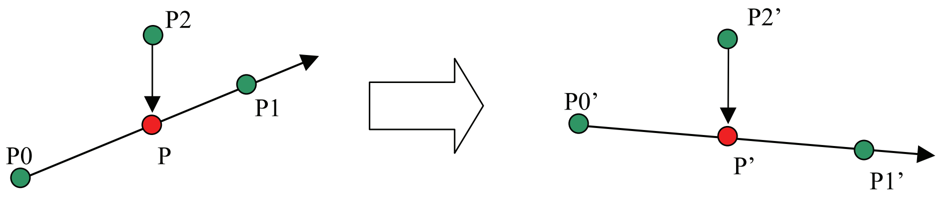


Fig. 6 *Perpendicular references with three points*

These dependencies may be defined and visualized in a graphic environment because the needed complexity may be obtained by nesting those constructs, but the basic rules are quite simple.

The mixture of those constructions with the definition of relative or absolute coordinates given by parameters is not easy, because the order of evaluation may be of importance.

5.2 External Reference Geometry

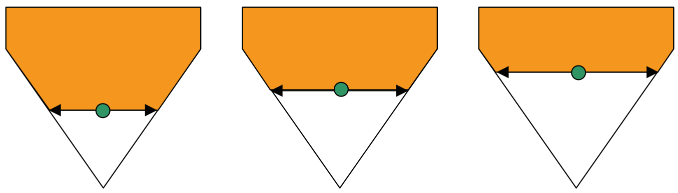


Fig. 7 *Perpendicular references with external reference axis*

Sometimes there is a dedicated axis available in the CAD-System defining either a point of the section explicitly or defining a point to be used in the above constructions. This allows a wide range of geometric constructions.

As an special example, we may take the location of a tendon which will be calculated in an external program to account for optimal vertical location, but the tendon has to follow the direction of the inclined web within the section. So the horizontal coordinate is the result of a function depending on the height itself. If we consider a curved web, then some reinforcements have to be taken as an offset to a variable depending on the axis. This examples shows clearly that we need a nested parameter evaluation in many cases.

5.3 Polar Coordinates and Circles

Fig. 8 *Polar reference and Radius reference*

Every single coordinate may be expressed by a formula using one or more parameters. Sometimes a direct specification of an angle and a polar distance is advantageous, similar to that of chapter 5.1, just replacing the second point by an angle.

On the other side a reference point may also define a circle by its location relative to the centre of the circle.

6. Examples

It is of course not possible to give a complex example in detail within a rather short paper. So I will first give an outline of the workflow and discuss then some finished examples.

The first step is the definition of the reference axis and all associated geometry. Then we have to define the parameters or variables along the axis. After this step, we have distinct values of our parameters at every selected parameter value s along the reference axis.

The second step is the definition of a master section. This section has to contain all geometric constraints along the bridge based on the elements “Coordinates by parameters”, “Coordinates by geometric reference” and mixtures of both. We need a formal distinction between formulas to be evaluated immediately and those which should be evaluated later. It is very helpful to have a quick slider allowing the user to see the variation of the section along the intended axis.

Now the structural system is generated by adding supports to the axis, and from that the finite element system is created. Every element has to have information about its ancestors, i.e. every beam element has the true curvature of the line available which has been used to generate it.

After having generated the locations of the sections, the sections are created by applying the defined rules of the master section for every instance along the reference axis. During evaluation we have now three sets of parameter scopes. The highest definition are the values of the parameters assigned to the axis, followed by those which are valid for the structure itself, followed by the default values of the master section itself.

6.1 Simple geometries

There are bridges which are very well suited to parametric modelling based on the reference axis. These have such an axis, and a varying geometry along that axis:

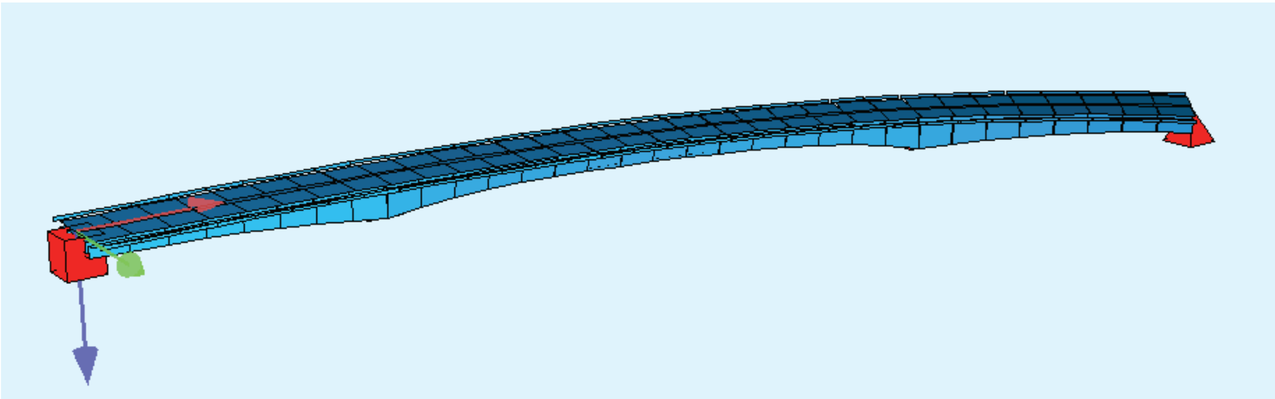


Fig. 9 *Curved Bridge with variant height*

Fig. 10 shows a system where not only the variant true geometry but also the effective width of the bridge deck may be easily described with variables along the reference axis:

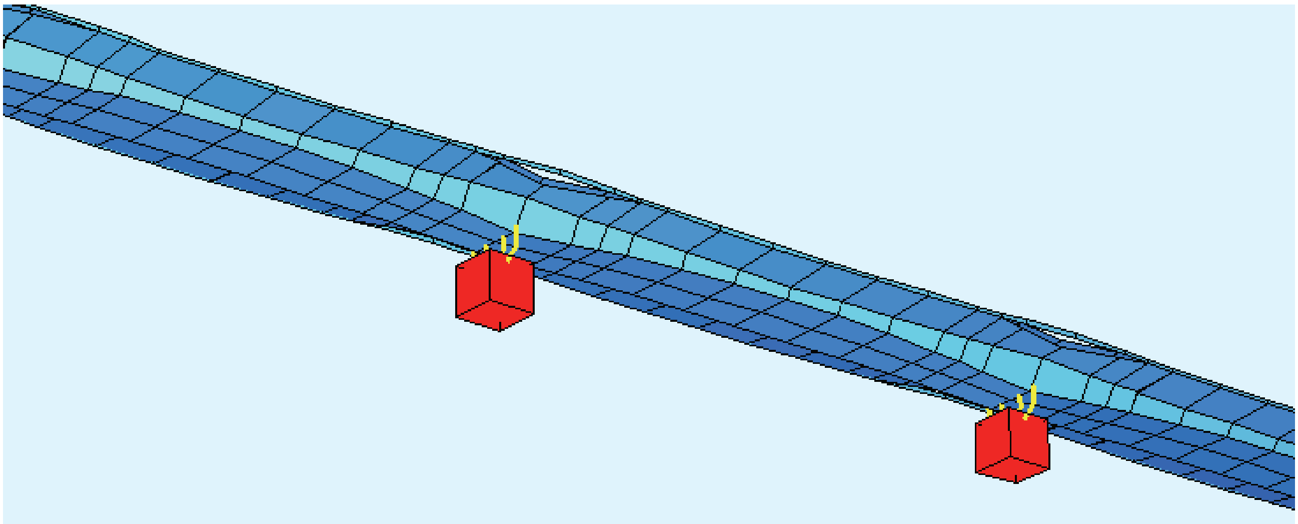


Fig. 10 *Non effective widths of a bridge*

6.2 Regular Structures

Quite often, we have bridges with a regular structure which can make use of repeatable instances of geometric elements like the following. Parametric modelling is introduced at least by the curved bridge axis. Additional structures may be connected to that model by defining the placement where such an element is connected to the main structure. But the main parametric information is that the number of elements depends on the total length of the structure and the size and the distance of the substructure elements.

However, any further data like loading should not rely on a particular system or span dimension. To allow this we have to use either very sophisticated macro structures or use brute force and many load cases or use influence lines and a software allowing to evaluate the live load depending on this enhanced data.

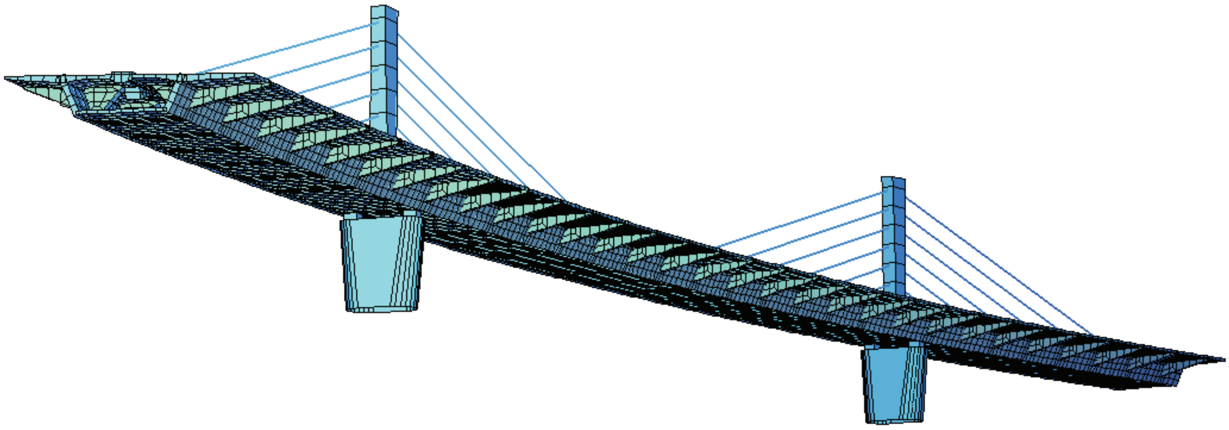


Fig. 11 *Bridge with regular structures*

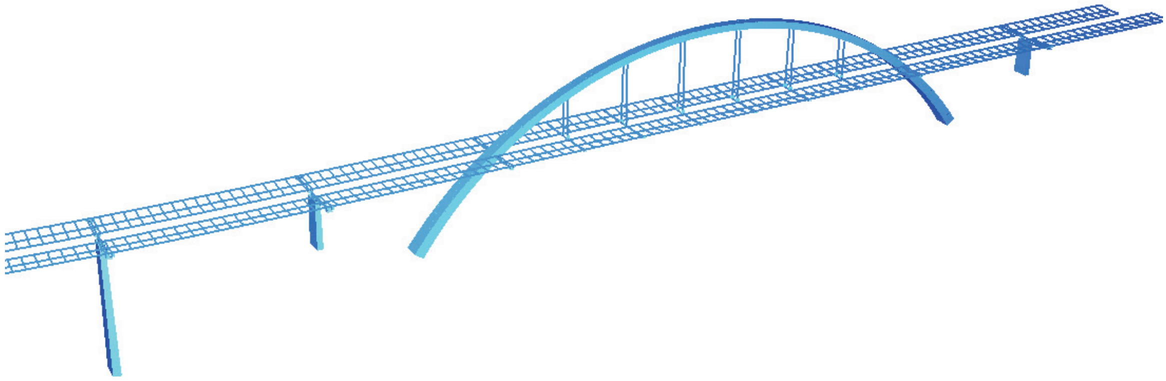


Fig. 12 *Bridge with regular structures (Swinesund)*

6.3 Integral Structures

There are also bridges which do not fit into the concept of a one dimensional reference axis like the following integral bridge:

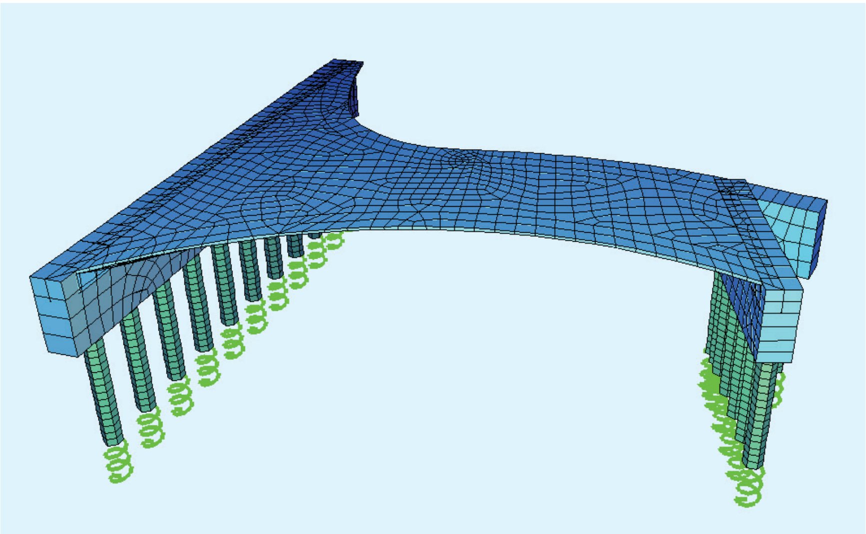


Fig. 13 *Integral bridge with multiple reference-axes*

6.4 And beyond

There are bridges where people are happy if they can manage to define the system with classical techniques, like the following example of the Zaragoza bridge pavillon for the EXPO 2008.

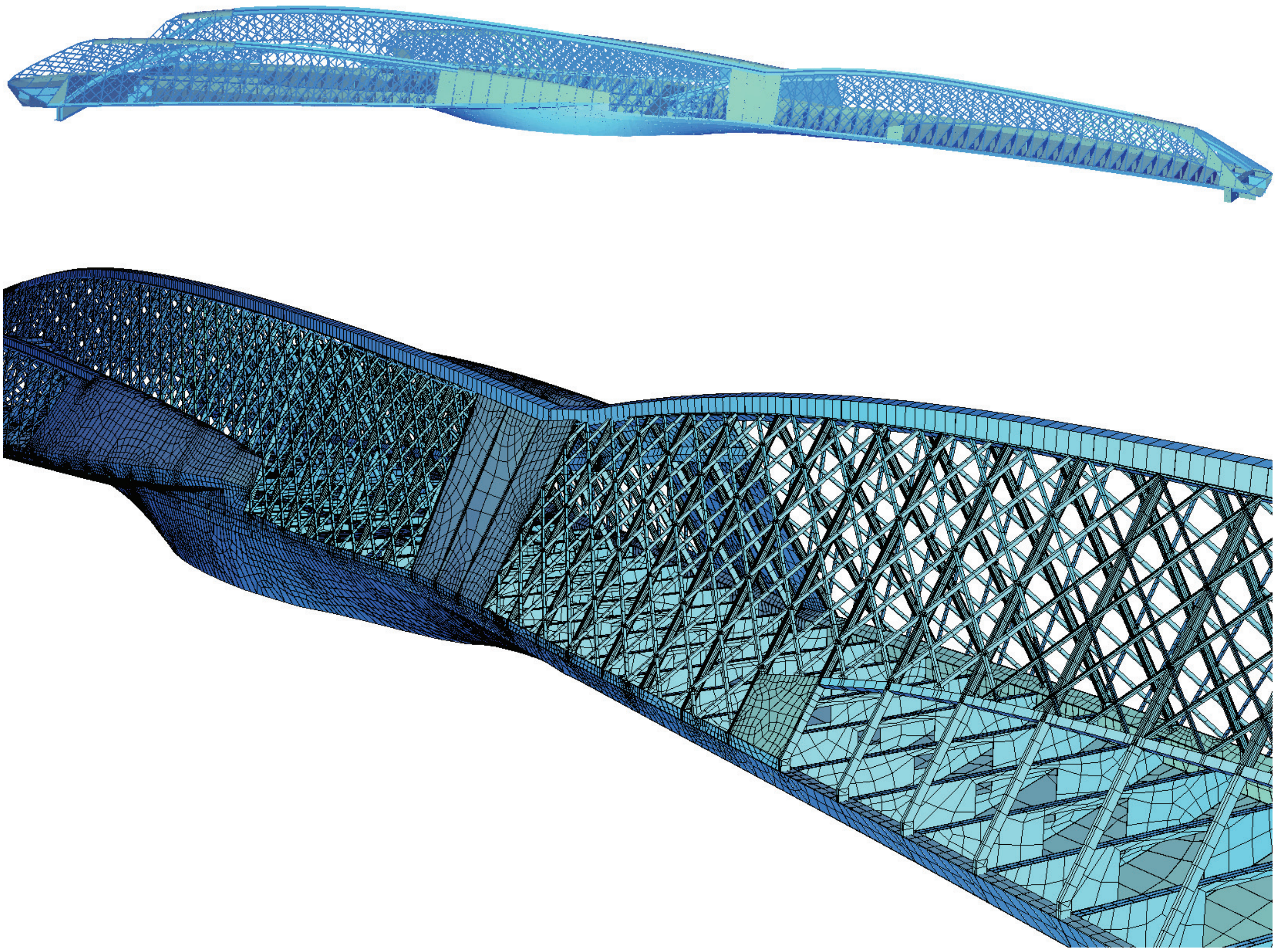


Fig. 14 *Zaragoza bridge pavillon*

7. Conclusions

There is advertisement for software allowing the user to design a complete bridge within a few hours. If this software is believed to be reliable, it can address only standard cases. For non standard cases some considerable effort must be invested in the modelling of the bridge.

Sometimes the best choice is to introduce the parameters only when it becomes evident that those parameters will change; but a good principle of software development is to anticipate changes unless the parameter has a pure mathematical meaning which can not change [4]. A recent research project in Bavaria [5] has the subject of creating a “virtual construction site” with interfaces between all engineering and construction tasks for bridge and road construction. One important aspect there is to use parametric modelling as much as possible to allow to evaluate the impact of changes.

Keeping in mind that a lot of information is lost during data processing, it is important to preserve original information as much as possible by using parametric modelling wherever possible. One important benefit of parametric modelling is that the software has always access to the exact data.

Especially for pre- or post-stressed concrete, it is well known that the design criteria are based on the difference of large numbers, and thus sensitive to small changes anywhere in the analysis process.

For the modelling of the bridge, the best solution should be a mixture of both approaches presented. If we start with a constructive model based on a reference axis, we are closer to the real nature of a bridge. But we can connect the 3D solid model with all its geometry based on coordinates deducted from the parameter definitions along the reference axis and then we have all we need to do a sound design, analysis, construction and facility management.

If it comes to the point where we have to exchange data, the common rule is that we lose all parametric definitions. If we want to exchange data, we should agree on some basic rules that may be either transferred or mapped to another description.

8. References

[1] AHN M., BÖCKELER K.H., HAAS W., Eingabekonventionen für CAD−Programme, CAD−Bericht Kfk−CAD 39, Kernforschungszentrum Karlsruhe, 1976

[2] SOFiSTiK Manual „General Description“ , SOFiSTiK AG, Oberschleißheim (1987-2007)

[3] LEBEC, E., IFC-Bridge V2 Data Model, IAI Project CI-2, 2005.

[4] HUNT A., THOMAS D., The pragmatic Programmer – From Journeyman to Master, Addison Wesley Longman, Inc.

[5] www.forbau.de