A New Program for the Design of Cable-Stayed Bridges

Edoardo ANDERHEGGEN
Professor
Swiss Federal Institute of Technology (ETH)
Zurich, Switzerland

Edoardo Anderheggen has been Professor of Computational Methods in Engineering since 1976 at the ETH.

Pietro PEDROZZI
Civil Engineer
Swiss Federal Institute of Technology (ETH)
Zurich, Switzerland

Pietro Pedrozzi, born 1975, graduated in civil engineering in 1999 at the ETH and is now working at his Ph.D. thesis at Prof. Anderheggen's institute.

Summary

A new finite element program has been developed which allows to easily simulate all construction stages of a cable-stayed bridge taking construction history dependent phenomena such as locked-in displacements, creep and shrinkage into account. The program also automatically computes all needed pre-camber values and the cables post-tensionings. It can also be used during the bridge erection to correct unexpected deviations from the planned geometry.

Keywords: Cable-stayed bridges; construction history; pre-camber; post-tensioning; locked-in displacements; creep; shrinkage; user friendliness.

1. Introduction

In cable-stayed bridges the deck is supported at more or less regular distances by cables which are fixed to the top or along a mast protruding from the deck plane. In most cases cable-stayed bridges are self-anchored, i.e. the normal force introduced in the deck by the cables on one mast's side is compensated by the normal force introduced on the other side.

The main advantages of cable-stayed bridges are that they can be built in very large spans (today with a central span of up to 1000 meters) by free cantilevering (see figure 1), offer a great stiffness, need little material and can look quite elegant.

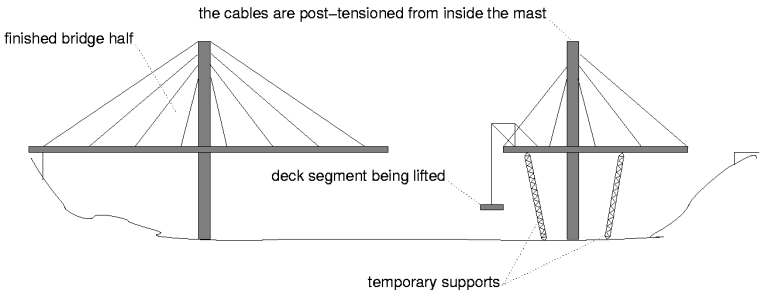

Fig. 1 Bridge erection by free cantilevering

Despite these advantages, cable-stayed bridges began being diffused only in the fifties because it was impossible to analyze them with a reasonable effort and a satisfactory accuracy using the manual methods of the pre-computer days.

The reason why the design and erection of cable-stayed bridges is so computationally intensive is that they are statically over-determined to a degree which is approximatively as large as the number of cables and, at the same time that their displacements and section forces are adjustable through the post-tensioning of each single cable.

In addition, long span cable-stayed bridges underlay non-linear phenomena such as cable sag and concrete creep.

The goal of adjusting the post-tensioning force in each cable and of mounting pre-cambered mast and deck segments is that of having a finished structure which reflects the planned geometry. Even though cable-stayed bridges are globally very stiff, single members can be quite flexible and produce large displacements, especially during construction.

Once statical analysis for several construction stages and for the final service state has been carried out, computations might still be needed during erection in combination with measurements in situ so as to correct alignment errors and to keep the rather unpredictable behaviour of concrete and supports under close control.

2. Use of standard structural analysis software

Cable-stayed bridges can, in principle be successfully analyzed with standard structural analysis software. This is however not straightforward. Standard software, in general, cannot automatically find the post-tensioning forces needed in the cable-stays or the pre-camber values for the mast and deck segments. In many cases these still have to be found by the bridge designer by means of time consuming and error prone manual computation. Here there are a few possible strategies:

- Insertion of fictitious hinges in the mast so as to find cable forces with which the mast is not submitted to bending moments.
- Use of very high artificial stiffnesses for the normal force, to avoid axial displacements in the deck and the pylons.
- Use of supports instead of cables to retrieve the post-tensioning forces from the support reactions.
- Determination of the bridge's displaced shape caused by a unitary shortening of every cable and a unitary pre-camber value for every deck and mast segment. Formulation of an equation system (to be solved in an extern program) representing as many compatibility conditions as unitary shortenings and pre-camber values, allowing to find the actually needed shortenings and pre-camber values (this is the method adopted in our program BRIDE but implemented in a totally automatic way).

These computations, when performed either manually or using additional external programs, are very time consuming and inhibit the investigation of several project variants.

3. The program BRIDE and its peculiarities

The difficulties encountered in the analysis of cable-stayed bridges with standard software and the opportunity offered by the technical progress in computer hard- and software appeared to be two good reasons to start the development of a computer program specially tailored on the requirements of cable-stayed bridge designers.
The fact that a growing number of programs for the analysis of cable stayed bridges is being offered shows that times are mature for fundamental research on this topic: this can provide a knowledge base for further research on technically newly accessible domains, asses commercially available programs or as a seed development for new commercial products.
The program has been called "BRIDE", an acronym for BRIdge DEsigner (it is a tradition of our institute to give programs romantic names), and its goal is to help the designer in the following tasks:

- model all kinds of cable-stayed bridges, taking all main non linear effects into account in a consistent way,
- easily allow the analysis of each construction stage,
- find all needed post-tensioning forces for the cables and all pre-camber values for the mast and deck segments in a fully automatic way and
- correct deviations in the geometry measured in situ by changing the cable forces.

Five main ideas determined the design of the program BRIDE and distinguish it from others programs:

- As the displacements are assumed to be small, which is acceptable being cable-stayed bridges very stiff, the behaviour of every structure's segment modelled with a finite element can be fully described with the element's local stiffness matrix plus a set of "initial"

displacements. Such an element does not depend on the global structure and, fully representing the physics of the modelled structure's segment, can be added to or removed from the numerical model just like the corresponding structure's segment can be added or removed in real life (see section 4).

- Post-tensioning forces or pre-camber values which have to be found automatically by the program can be formulated as initial element displacements whose magnitudes are found through conditions to be fulfilled (see section 5).
- The erection of a cable-stayed bridge is an evolutionary process and the simplest way to describe it with accuracy is a list of the changes occurring in the system in chronological order (a "chronological" objects list, see section 6).
- In order to take displacements locked in concrete and creep into account it is necessary to know the displacements of the structure in the stages prior to the stage being analyzed (see section 7).
- The program BRIDE should be user friendly enough to be applicable both in the bridge construction industry and for teaching purposes.

4. Element description through its local stiffness matrix and a set of initial displacements

The geometry of space frame finite element models such as those analyzed by the program BRIDE is described through the Cartesian coordinates of the nodes and through the element's incidence nodes. In addition, every element requires section and material properties. Once the Cartesian coordinates of the incidence nodes, the material and the section of the element are defined, its elastic response can be formulated by means of a local 12x12 elastic stiffness matrix.

If we would consider an existing bridge already modelled in a simulation program (see figure 2), take a concrete cutter, cut out a segment of the structure modelled with a single finite element (see figure 3) and determine its geometry in the unloaded state, we would notice that its geometry does not correspond to the geometry of the finite element. In fact, displacements “locked-in” during the hardening of the concrete, creep, shrinkage and the pre-camber would have slightly changed the shape of the cut out, thus unloaded segment (see figure 4).

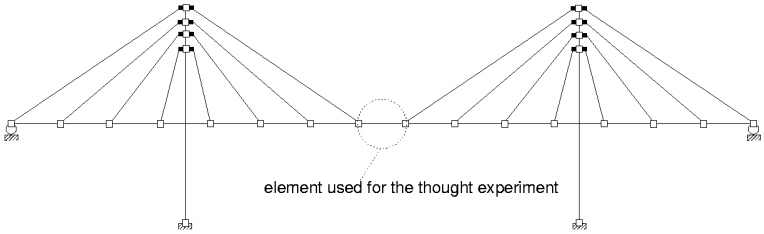

Fig. 2 Step 1. Consider an element in the numeric model of an existing bridge

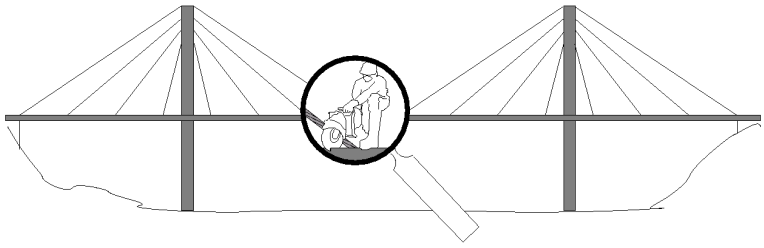

Fig. 3 Step 2. Cut out a structure segment corresponding to a finite element of an existing bridge

The vector $\mathbf{a}_{initial}$ of size 12 (3 translation and 3 rotation components for each incidence node) containing the difference between the cut out segment's and the finite element's geometry can be formulated as follows:

$$\mathbf{a}_{initial} = \mathbf{a}_h + \mathbf{a}_{cr} + \mathbf{a}_{sh} \qquad (1)$$

where $\mathbf{a}_h$ is the contribution of the concrete locked-in displacements, $\mathbf{a}_{cr}$ that of creep and $\mathbf{a}_{sh}$ is that of shrinkage (the pre-camber is not taken into account here because it is considered as a "conditional" load, see section 5).

One way to model the cut segment correctly is to reproduce its displacements by submitting the finite element to a set of initial self-equilibrating forces **f** (see figure 5) producing the initial

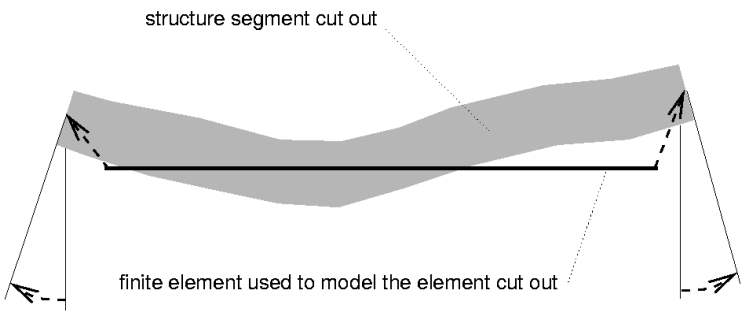

Fig. 4 Step 3. The geometry of the structure segment cut out and the geometry of the finite element used to model it differ by the initial displacements $\mathbf{a}_{initial}$

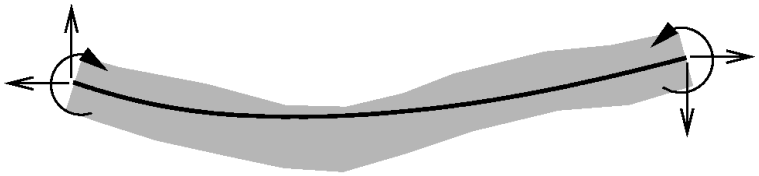

Fig. 5 Step 4. The finite element submitted to the initial self-equilibrating force **f** *assumes the same geometry as the structure segment cut out*

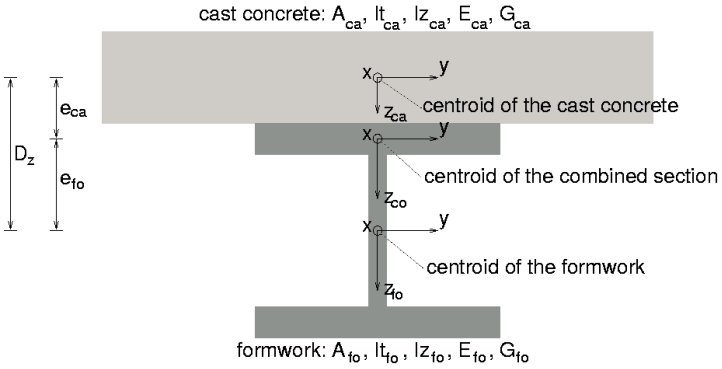

Fig. 6 Section idealization used for composite beam elements

displacements $\mathbf{a}_{initial}$. The forces **f** are found by multiplying the element local stiffness matrix **k** with $\mathbf{a}_{initial}$:

$$\mathbf{f} = \mathbf{k} \cdot \mathbf{a}_{initial} \qquad (2)$$

The components of the vector **f**, like for any kind of element load, are to be assembled in the global load vector every time the element is part of the model, its local stiffness matrix **k** being assembled in the global stiffness matrix.

Being initial, i.e. corresponding to a deformed but stress-free state, the contribution from **f** to the section forces has to be ignored during post processing, i.e. the section forces induced by **f** in the element have to be subtracted from the section forces found from the nodal displacements of the global solution. This subtraction does not affect the equilibrium in the structure because **f** is self-equilibrating: this is a property of local stiffness matrices which, being singular, always deliver self-equilibrating forces group if multiplied with any nodal displacement vector.

Initial displacements have to be taken into account in a slightly different way for the composite beam element specially developed for the program BRIDE, which allows to take the bond between concrete and its underlying steel profile into account (see figure 6). The composite beam element takes bond into account by merging into a single combined section the section of the cast concrete and the section of the steel profile, which is modelled with a so-called formwork element. For composite beam elements **f** is found as follows:

$$\mathbf{f} = \mathbf{k}_{co} \cdot \mathbf{a}_h^{co} - \mathbf{q}_{co}^{fo} + \mathbf{k}_{ca} \cdot (\mathbf{a}_{cr}^{ca} + \mathbf{a}_{sh}^{ca}) + \mathbf{k}_{fo} \cdot (\mathbf{a}_{cr}^{fo} + \mathbf{a}_{sh}^{fo}) \qquad (3)$$

where $\mathbf{k}_{co}$ is the stiffness matrix computed with the combined section, $\mathbf{a}_h^{co}$ and $\mathbf{q}_{co}^{fo}$ are the displacements and the nodal forces of the formwork beam during the casting stage, $\mathbf{k}_{ca}$ and $\mathbf{k}_{fo}$ are the stiffness matrices of the cast concrete part and of the formwork beam, respectively, with no bond between them. $\mathbf{a}_{cr}^{ca}$, $\mathbf{a}_{sh}^{ca}$, $\mathbf{a}_{cr}^{fo}$ and $\mathbf{a}_{sh}^{fo}$ are the initial displacements for the formwork beam and for the cast concrete beam (it is justified to have also initial displacements due to creep and shrinkage for the formwork since this might be in pre-cast concrete). All this forces, stiffness matrices and displacements are expressed relatively to the centroid of the combined section. The element is then excentrically connected to its nodes.

The fact of not taking bond into account for $\mathbf{k}_{ca}$ and $\mathbf{k}_{fo}$ is a conspicuous simplification which is acceptable because the self-equilibrating load **f** is applied on the un-displaced composite element (whose stiffness takes bond into account) hence differences between the initial displacements of the cast concrete $\mathbf{a}_{cr}^{ca} + \mathbf{a}_{sh}^{ca}$ and those of the formwork $\mathbf{a}_{cr}^{fo} + \mathbf{a}_{sh}^{fo}$ lead just to differences in the section

forces but not to bond producing discontinuities in the total strain (the elastic and the initial strain summed together) because in the finite element model the element is “forced” in its un-displaced shape for the formulation of global equilibrium conditions the elastic strain must compensate any initial strain.

5. Conditional loads

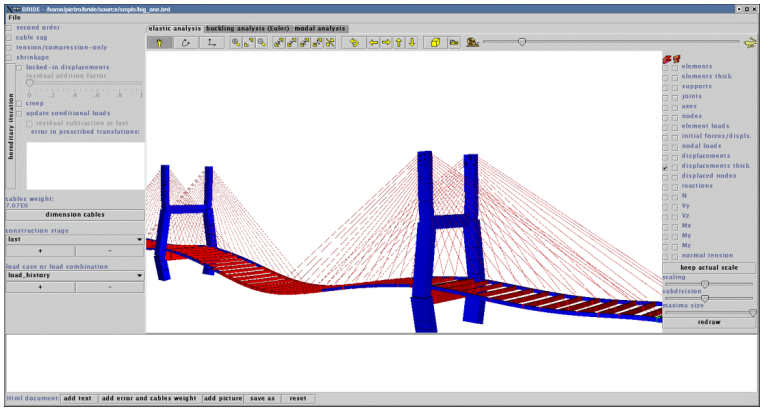

Fig. 7 Snapshot of the program BRIDE showing the deflected shape of a model in which the cables have not been post-tensioned

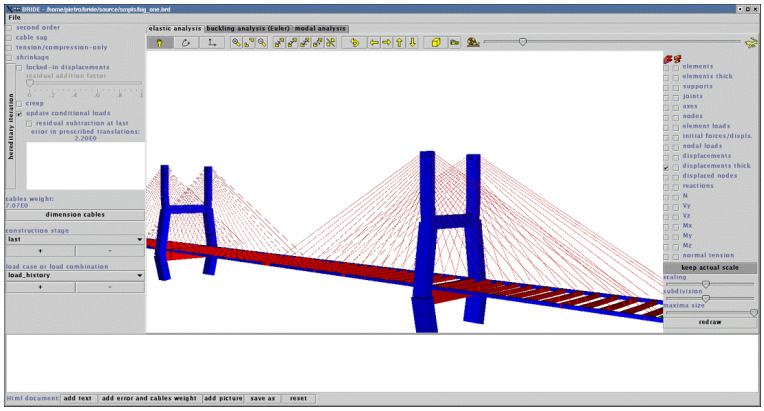

Fig. 8 Snapshot of the program BRIDE showing the deflected shape of the same model as in figure 8 but post-tensioned with automatically found intensities

The so-called conditional loads represent the post-tensioning of the cable-stays and the pre-camber of mast and deck segments. Their intensities are automatically computed by the program BRIDE.

Both conditional loads and regular loads are model objects to be inserted in the chronological objects list (see section 6). The only differences is that in regular loads the load intensity is explicitly defined while in conditional loads it is defined through a condition to be fulfilled, and that conditional loads always belong to the standard load case *load_history* (see section 7). Their needed intensities are found automatically by the program BRIDE by fulfilling appropriate conditions specified by the program user (e.g. "post-tension a cable with such a force that its anchoring point on the deck has no vertical displacement in the last construction stage, taking into account the dead load and all other conditional loads", see figures 7 and 8).

6. Chronological objects list: a full description of the erection process

After some attempts it became clear that the most convenient way to specify the evolving model of a cable-stayed bridge during construction is a list of all model's objects (nodes, elements, supports, forces, ...) ordered in the same chronological order in which their real counterparts are added to the bridge being constructed. The object "stage" has been introduced to allow the program user to define a new construction stage whenever he feels that enough model objects have been added to the model since the last stage object. The stage objects define the states of the bridge where analysis is possible.

Once the program BRIDE has read the objects list defined by the user, it constructs a list of data objects representing it.

To build the numerical model of any construction stage, the (“object-oriented”) program BRIDE goes through the list of data objects from its beginning, sends to each data object the signal to assemble himself to the actual model and stops when it encounters the data object corresponding to

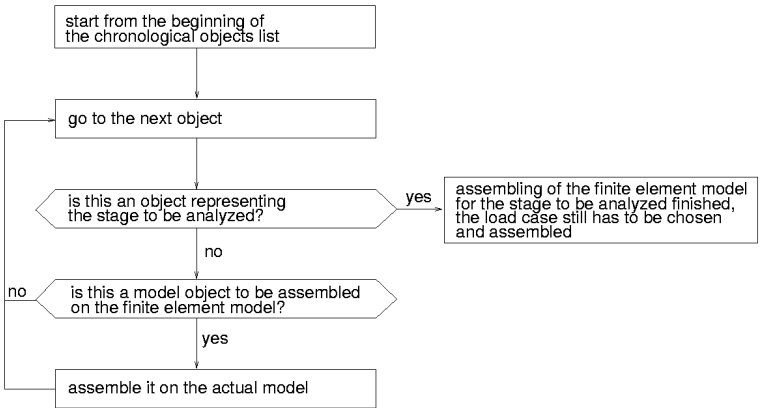

Fig. 9 Flow diagram of the assembling of a construction stage

the desired construction stage (see figure 9).

To analyze the model for a given load case the program BRIDE proceeds in an analogous way as for the assembling of the finite element model: it goes through the chronological objects list from the beginning, takes into account the load objects belonging to the chosen load case and stops when the stage object corresponding to the stage being analyzed is encountered.

7. Hereditary iteration: an automatic analysis of each construction stage in chronological order

To take creep and locked-in concrete displacements in a given construction stage into account it is necessary to know the initial displacements $\mathbf{a}_{cr}$ and $\mathbf{a}_{sh}$ for every element (see section 4), which, in turn, depend on the displaced shapes of the bridge in all previous construction stages. Since it is important to allow the user to view the results of any construction stage immediately, without having to wait for the analysis of all former stages, the nodal displacements of every construction stage due to all relevant loads are stored on the corresponding stage data object.

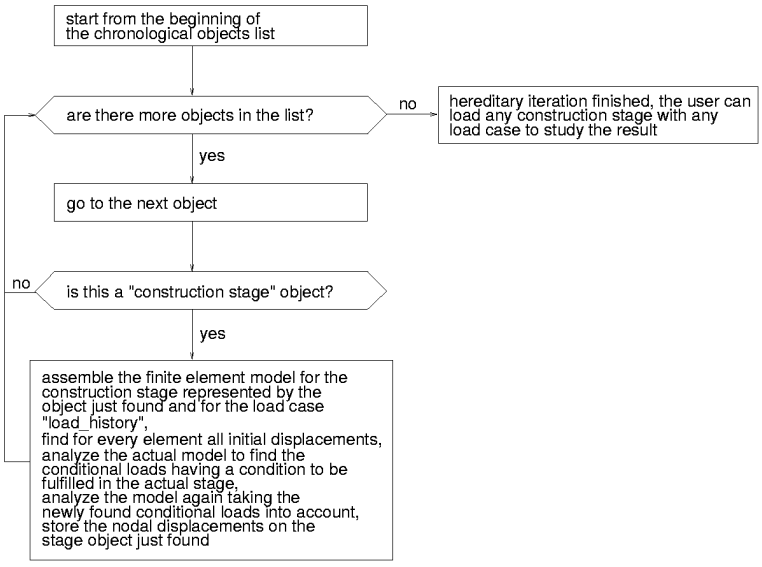

Fig. 10 Flow diagram of the hereditary iteration

Such nodal displacements are found during what is called “hereditary iteration”, in which every construction stage is automatically assembled and analyzed in chronological order. Figure 10 shows the flow diagram of such iteration. To denote the load objects which should be taken into account during the hereditary iteration the standard load case called *load_history* has been introduced. The conditional loads are automatically assigned to this standard load case because they are to be computed during the hereditary iteration. The user, however, is still free to assign them to other load cases or load combinations.

8. Conclusions

The program BRIDE has been presented which allows to take the linear and non-linear behaviour of cable-stayed bridges into account in a natural and accurate way during both initial design and construction. Different original approaches have been implemented in the program. The algorithms specially developed for the program BRIDE allowing to take such non-linear effects into account are briefly explained here.