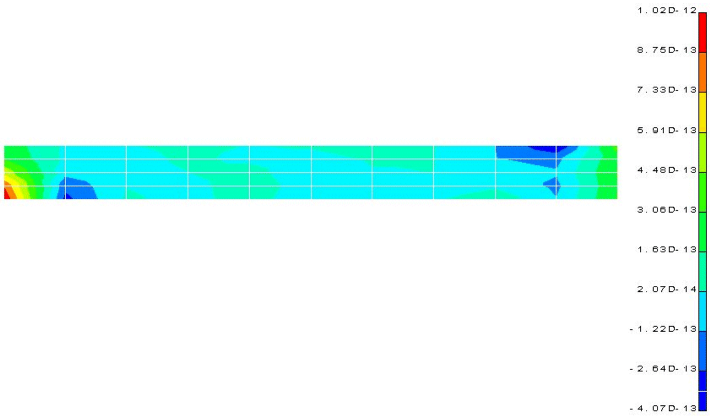

Figure 3. Vertical stress contour when the constraint distribution has been modified. Now the vertical stress value along all the domain is practically null. Its order of magnitude is 10^{-12}.

It must be said that generally calculation codes can give, on demand, the reaction forces values in the constrained d.o.f.; this information is very precious, both for joints calculation (as we mentioned in Chapter 2) and to check the correctness of load application: in fact commercial codes generally give also the resultant vector of the reaction forces which, together with the resultant vector of the applied loads, makes this check possible. If the two resultants are different (a part, clearly, from small numerical differences) it means that some forces or moments have been “lost” during the solution, due to numerical errors; we will come back to this aspect in Chapter 7, when we will discuss about models validation methods.

The last aspect we just mention is about unilateral constraints: a typical example is constituted by simple supports. Unfortunately a “simple” way to create unilateral constraints does not exist; in fact general constraints act on the d.o.f. in such a way to make it “non active”, while a unilateral constraint should maintain a certain d.o.f. “active” in one direction and “non active” in the opposite direction. This approach requires iterative techniques, which are based on non linear calculations.

3.3 Load conditions

Load conditions are the dual case of constraint conditions; in fact where displacements are known (i.e. constrained grids) the reaction forces are unknown), while where forces are known (i.e. loaded grids) the displacements are unknown.

It is practically impossible to generate numerical errors when applying load conditions, in opposition to what can happen with constraint conditions. The only mistakes that can be done are the conceptual ones; for example, as for constraints, applying a concentrated moment to grids belonging to solid elements is wrong (and must not be taken for granted that the code in use will produce a warning message), as we already discussed above.

Another example that apparently gives some unexpected results is the application of a force divided by the number of grids on the edge where the load has to be applied. Let's take once again the beam shown in figure 1 and let's load it by an axial force F = 1000 N. Let's discretize the vertical edge in four elements (and therefore five grids); the first thing that one may think is to apply to each grid a force f = 200 N. Figure 4 shows the deformed shape of the beam in the region where the load has been applied following this approach; it can be noted that the grids at the vertices deform more than the others belonging to the same vertical edge. This is due to the fact that in the vertices there is only one element to react the applied force f, while in the other location two elements are present. In order to avoid this problem, to the vertex grids half the force (with respect to the other "internal" grids) should be applied, taking into account that the total force F is still equal to 1000 N. In this case we will have:

3 central grids = 250 N each
2 vertex grids = 125 N each

The total is 250 x 3 + 125 x 2 = 1000 N.

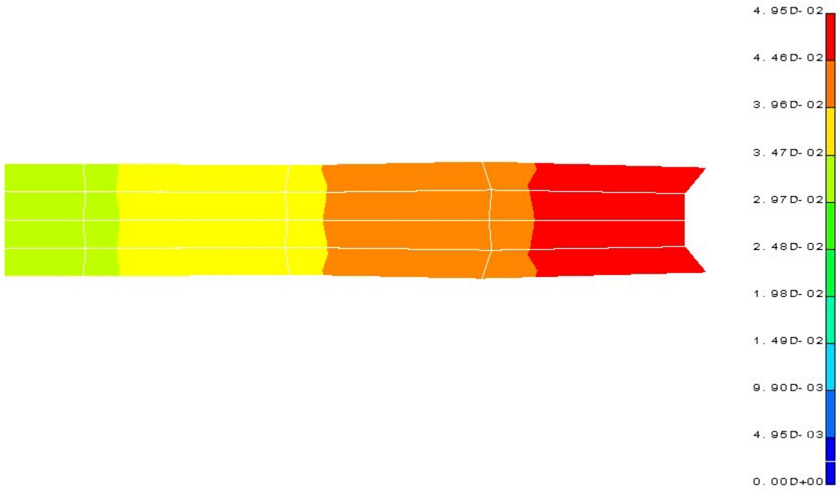

Figure 4. Longitudinal displacement contour. The deformed shape looks "strange" for a pure axial load.

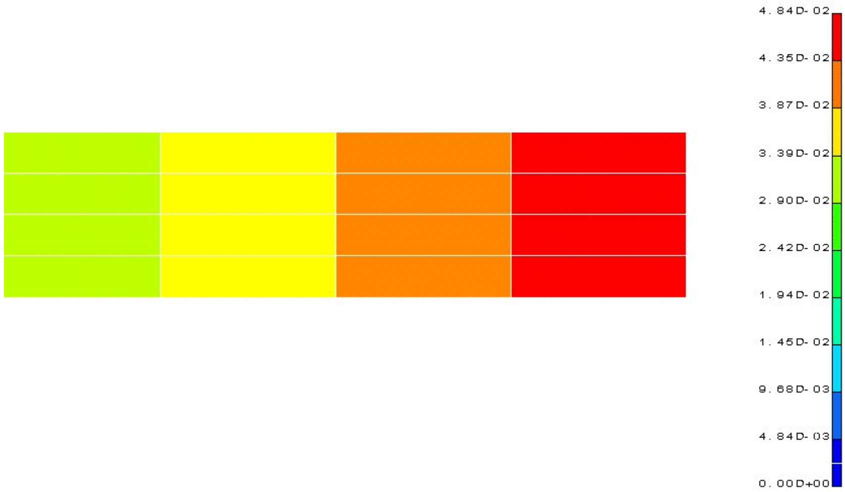

Figure 5. Longitudinal displacement contour. Now the deformed shape appears congruent to what can be foreseen.

Figure 5 shows the deformed shape of the beam when the new nodal load distribution has been applied. One can say that this error is more "esthetical" than substantial; this can be true in part, but frequently it is necessary to know with good accuracy the value of the displacement in a given point of a certain structure, also in a region where the loads are applied. Moreover also the stress distribution is altered in these zones. It is therefore necessary to use the best possibilities that the finite element method offers, taking care also of these details.

As above discusses that load conditions are the dual case of constraint conditions; however for loads an extension exists. In fact very often it is necessary to apply pressure loads; this is done by selecting element faces, instead of grids when applying forces and moments. It is the calculation code that automatically converts this information and recalculates the suitable forces to be applied to the grids. For this case the most common mistake is made when using shell elements, when the normals of the elements involved in the pressure load are not oriented homogeneously. In fact the calculation code, having to manage an element that does not have a "physical" thickness, cannot establish in which direction the material is present (while it can perform this operation easily when it deals with solid elements); in order to decide which will be the positive value for the pressure load the software "looks" at the normal orientation.