An explanation on how to compute deflections of a statically indeterminate beam

A primer on deflections of statically indeterminate structures

Authors:
S.J. van Elsloo
Dr. ir. J.M.J.F. van Campen
Dr. ir. W. van der Wal

AE3212-II
Simulation, Verification & Validation
February 2021

Deflections of statically indeterminate structures

This document aims to provide a primer on the computation of deflections of statically indeterminate structures, including how to derive the shear force, bending moment and torque diagrams.

Sample load case

In order to illustrate the theory discussed in this document, an example of a statically indeterminate beam is shown in Figure 1. The beam is a simple rectangular beam with width C, height h and length L. The coordinate system used is shown in red; it is located at the 'leading edge' of the beam, at the midpoint of the front edge. The beam is subjected to a distributed load $w(x, z)$ that acts perpendicular to the xz-plane, in negative y-direction. At $x = x_1$, a point load P acts on the beam, applied in negative z-direction, at the midpoint of the front edge. The boundary conditions are as follows ($v(x)$ denotes the deflection in positive y-direction; $w(x)$ denotes the deflection in positive z-direction, and $\theta(x)$ denotes the twist angle around the x-axis, leading edge downwards is positive):

- The root of the beam is fixed, such that $v(0) = 0$, $v'(0) = 0$, $w(0) = 0$, $w'(0) = 0$ and $\theta(0) = 0$.
- At $x = x_2$, the top of the front edge is held in place in vertical direction (the displacement in vertical direction is 0). It is free to move in z-direction, though.
- At $x = L$, the bottom of the front edge is lifted up a distance d_1 in vertical direction, and it is held in place in horizontal direction (the displacement in horizontal direction is 0).

Thus, the total number of boundary conditions is 8. It is self-evident that this results in a statically indeterminate structure.

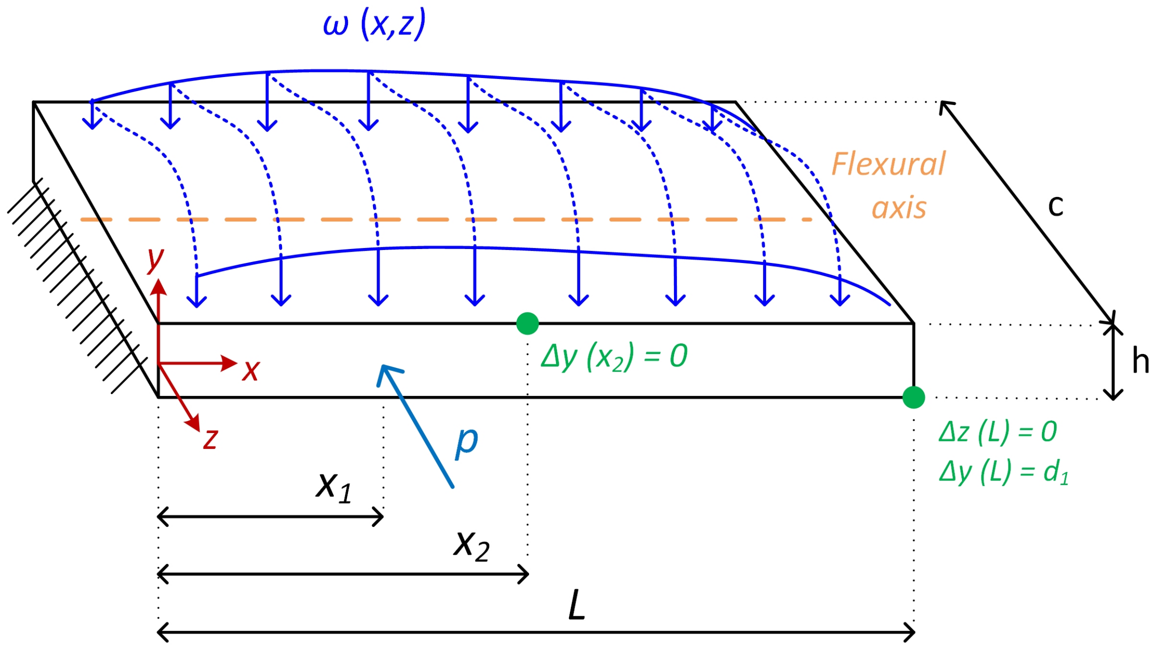

Figure 1: Sketch of an example load case acting on a cantilevered beam.

Free body diagram

Based on these boundary conditions, the free body diagram of the beam can be drawn, as shown in Figure 2. At the root, two reaction forces act (R_{1_y} and R_{1_z}; the axial force is ignored) and three couple moments: M_{1_x}, M_{1_y} and M_{1_z}. At point 2, there is a reaction force R_{2_y}; at point 3 there are reaction forces R_{3_y} and R_{3_z}. Each reaction force/moment has been assumed to be in the positive coordinate system and no attempt is made to predict which direction they will actually point in.

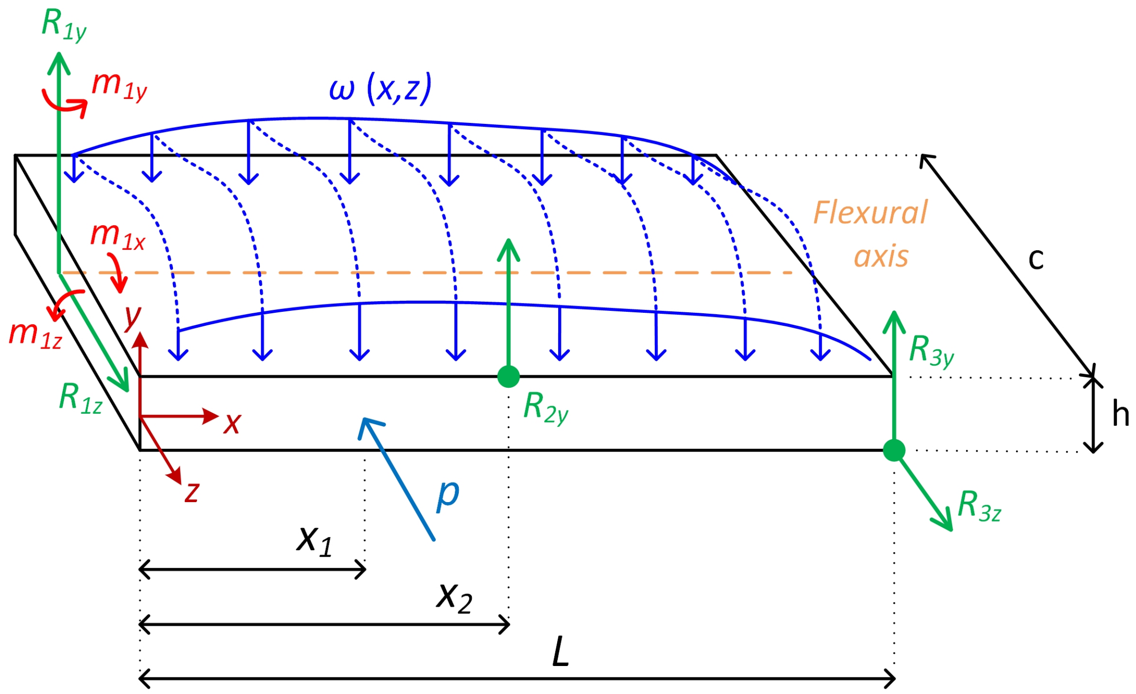

Figure 2: Free body diagram of the cantilevered beam.

Loading distribution functions

It is then necessary to set up functions describing the variation of the bending moment and torque over the span of the beam. The bending moment about the y-axis may be described by (using the convention that a positive bending moment should cause in the yz-plane where z is positive)

$$M_y(x) = -M_{1_y}[x]^0 - R_{1_z}[x]^1 + P\left[x - x_1\right]^1 - R_{3_z}[x - L]^1 \tag{1}$$

where $[x]$ denotes a Macaulay step function, which is equal to x if $x \geq 0$, but equal to 0 when $x < 0$. Furthermore, $[x]^0 = 0$ if $x < 0$ and $[x]^0 = 1$ if $x \geq 0$. Macaulay step functions may be differentiated and integrated as if they were a proper monomial. A point force has a Macaulay step function of order 1 associated to it; a couple moment has a Macaulay step function of order 0 associated with it.

In order to justify the signs, consider the following thought experiment: ignore all the existing boundary conditions, and assume only the end at $x = L$ to be cantilevered instead. Consider the effect of P on the bending moment: it will push the beam away, and will result in the front end of the beam to be under tension and the back end of the beam to be under compression, since we are currently assuming the end at $x = L$ to be cantilevered. Since the front end is in the direction of positive z, this means that P induces a positive bending moment in Equation (1). A similar reasoning holds for why R_{1_z} and R_{3_z} produce a negative bending moment (they are each assumed to point in the opposite direction of P, so it is only natural that they get the opposite sign of P).

For M_{1_y}, note that a couple moment is essentially two forces acting close to each other, but in opposite direction: the direction of M_{1_y} drawn would indicate that there is a point force pointing in negative z-direction just to the right of the end at $x = 0$, and a point force pointing in positive x-direction just to the left of the end. This would cause the beam to deflect towards the positive z-direction, and thus cause the back of the beam to be under tension and the front end under compression. Since the front end is in the direction of positive z, this means that M_{1_y} induces a negative bending moment in Equation (1).

Thus, to establish the signs, it may be helpful to consider the beam without any boundary conditions and to do a thought experiment with the end at $x = L$ to be cantilevered.

The bending moment about the z-axis is slightly more complicated, due to the presence of the distributed load. It is first necessary to integrate the distributed load over z, such that we obtain a distributed load that only depends on x; in other words, let

$$\bar{w}(x) = \int_{-C}^{0} w(x,z)\,dz \tag{2}$$

The bending moment is then given as function of x by

$$M_z(x) = M_{1_z}[x]^0 - R_{1_y}[x]^1 + \int_0^x \int_0^{x_a} \bar{w}(\tilde{x})\,d\tilde{x}\,dx_a - R_{2_y}\left[x - x_2\right]^1 - R_{3_y}[x - L]^1 \tag{3}$$

where signs may be established analogously to how they were found before.

Finally, the torque as function of x may be established. For this, it is important to realise that torque occurs around the shear center. Let us assume that the y-coordinate of the shear center is located at the middle of the beam (thus $\hat{y} = 0$), and let the z-coordinate be denoted by $\hat{z}$. Furthermore, define a leading edge downwards to be positive[1]. The distributed torque (in Nm/m) of the distributed load can be expressed as

$$\tau(x) = \int_{-C}^{0} w(x,z)(z - \hat{z})\,dz \tag{4}$$

The distributed torque is then equal to (with R_{3_z} acting a distance of $h/2$ below the shear center, and R_{2_y} and R_{3_y} acting a distance $0 - \hat{z}$ in front of the shear cente; note that $\hat{z}$ is negative so $0 - \hat{z}$ is positive)

$$T(x) = M_x[x]^0 + \int_0^x \tau(\tilde{x})d\tilde{x} - (0 - \hat{z})\,R_{2_y}\left[x - x_2\right]^0 - (0 - \hat{z})\,R_{3_y}[x - L]^0 - \frac{hR_{3_z}}{2}[x - L]^0 \tag{5}$$

Evidently, we have 8 unknown parameters: M_{1_x}, M_{1_y}, M_{1_z}, R_{1_y}, R_{1_z}, R_{2_y}, R_{3_y} and R_{3_z}. We thus want to obtain 8 equations in order to obtain a consistent system of equations. We will now explore how to obtain these equations.

Recovering the equilibrium equations

The first five equations may be straightforwardly found by recovering the equilibrium equation from the bending moment and torque distributions. First, we know that at $x = L$, we must have $M_y(L) = 0$, $M_z(L) = 0$ and $T(L) = 0$. Thus, plugging $x = L$ into Equation (1), (3) and (5) and setting the result equal to 0 results in three equations.

Not only the bending moment and torque should be 0 at $x = L$, also the shear forces should be zero. The shear forces may be found by differentiating the bending moments:

$$S_z(x) = \frac{dM_y}{dx} = -R_{1_z}[x]^0 + P\left[x - x_1\right]^0 - R_{3_x}[x - L]^0 \tag{6}$$

$$S_y(x) = \frac{dM_z}{dx} = -R_{1_y}[x]^0 + \int_0^x \bar{w}(\tilde{x})\,d\tilde{x} - R_{2_y}\left[x - x_2\right]^0 - R_{3_y}[x - L]^0 \tag{7}$$

Thus, plugging in $x = L$ into Equation (6) and (7) and setting the result equal to 0 results in two additional equations. It is noted that these five equations are essentially the equilibrium equations of the forces acting on the aileron.

[1]This is also consistent with the right-hand-rule around the drawn x-axis.

Using the moment curvature relationships

Nonetheless, the system of equations is still undetermined. One can now make use of the moment-curvature relations and the relation between torque and rate of twist to find the remaining equations necessary to solve this problem, as will be shown in this section.

First, it is noted that

$$\frac{d^2v}{dx^2} = -\frac{1}{EI_{zz}}M_z \tag{8}$$

Subsequently, we may integrate the above expression twice to obtain

$$\begin{aligned} v'(x) = &-\frac{1}{EI_{zz}}\Big(M_{1_z}[x]^1 - \frac{R_{1_z}}{2}[x]^2 + \int_0^x\int_0^{x_b}\int_0^{x_a}\bar{w}(\tilde{x})\,d\tilde{x}\,dx_a\,dx_b \\ &- \frac{R_{2_y}}{2}\left[x-x_2\right]^2 - \frac{R_{3_y}}{2}[x-L]^2\Big) + C_1 \end{aligned} \tag{9}$$

$$\begin{aligned} v(x) = &-\frac{1}{EI_{zz}}\Big(\frac{M_{1_z}}{2}[x]^2 - \frac{R_{1_z}}{6}[x]^3 + \int_0^x\int_0^{x_c}\int_0^{x_b}\int_0^{x_a}\bar{w}(\tilde{x})\,d\tilde{x}\,dx_a\,dx_b\,dx_c \\ &- \frac{R_{2_y}}{6}\left[x-x_2\right]^3 - \frac{R_{3_y}}{6}[x-L]^3\Big) + C_1x + C_2 \end{aligned} \tag{10}$$

Clearly, this introduces two additional unknowns[2], but it will be shown later that the boundary conditions provide a sufficient number of equations to also solve for these.

Similar to the above, we also have

$$\frac{d^2w}{dx^2} = -\frac{1}{EI_{yy}}M_y \tag{11}$$

and thus, integrating twice, we obtain

$$w'(x) = -\frac{1}{EI_{zz}}\left\{-M_{1_y}[x]^1 - \frac{R_{1_z}}{2}[x]^2 + \frac{P}{2}\left[x-x_1\right]^2 - \frac{R_{3_z}}{2}[x-L]^2\right\} + C_3 \tag{12}$$

$$w(x) = -\frac{1}{EI_{zz}}\left\{\frac{-M_{1_y}}{2}[x]^2 - \frac{R_{1_z}}{6}[x]^3 + \frac{P}{6}\left[x-x_1\right]^3 - \frac{R_{3_z}}{6}[x-L]^3\right\} + C_3x + C_4 \tag{13}$$

Finally, we have for the relation between the twist and torque

$$\frac{d\theta}{dx} = \frac{T}{GJ} \tag{14}$$

and thus, integrating, we otain

$$\theta(x) = \frac{1}{GJ}\left\{M_x[x]^1 + \int_0^x\int_0^{x_a}\tau(\tilde{x})\,d\tilde{x}dx_a - (0-\hat{z})R_{2_y}\left[x-x_2\right]^1 - (0-\hat{z})R_{3_y}[x-L]^1 - \frac{hR_{3_z}}{2}[x-L]^1\right\} + C_5 \tag{15}$$

[2] Note that in fact, $C_1 = v'(0)$ and $C_2 = v(0)$ (since we are evaluating a definite integral between $x = 0$ and $x = \tilde{x}$), and both are given as boundary conditions, so technically, they are not 'unknowns' for this example. However, to avoid a loss of generality we will treat C_1 and C_2 as unknowns in the following. The same applies for the integration constants for the other degrees of freedom.

Boundary conditions

The above integrations have so far not given us any additional equations to solve for the unknowns; to the contrary, it has only introduced five additional unknown integration constants, bringing the total number of unknowns to 13, whereas we have only five equations so far. However, there are also eight boundary conditions, which allow for an additional eight equations.

Implementing the boundary conditions at the root is easy: they are simply $v(0) = 0$, $v'(0) = 0$, $w(0) = 0$, $w'(0) = 0$ and $\theta(0) = 0$. Plugging in $x = 0$ and setting the result equal to zero in Equation (9), (10), (12), (13) and (15) provides five equations. The remaining two boundary conditions are more complicated. It may seem at first that the appropriate boundary condition is

$$v\left(x_2\right) = 0 \tag{16}$$

However, it should be noted that part of the vertical displacement of point 2 is caused by the twist of the beam around the shear center. Indeed, assuming the twist to be small, the vertical displacement of point 2 caused by the twist is equal to $\theta\,(\hat{z} - 0)$, since point 2 is located at $z = 0$. Consequently, the correct formulation of the boundary condition is

$$v\left(x_2\right) + \theta\left(x_2\right)(\hat{z} - 0) = 0 \tag{17}$$

Similarly, the fact that point 3 is displaced a distance d_1 results in the boundary condition

$$v\,(L) + \theta\,(L)\,(\hat{z} - 0) = d_1 \tag{18}$$

Finally, the fact that point 3 is fixed in z-direction results in the boundary condition

$$w\,(L) - \theta\,(L) \cdot \frac{h}{2} = 0 \tag{19}$$

since the twist introduces a displacement in z-direction equal to $-\theta(L) \cdot h/2$, as point 3 is located $h/2$ below the flexural axis.

Conclusion

In the preceding notes, we have obtained a linear set of 13 equations for 13 unknowns. This may be straightforwardly solved using a matrix equation. From this, all reaction forces and moments are determined, meaning that the shear force, bending load and torsion as function of x are known (following Equation (1), (3), (5), (6) and (7)). Furthermore, with the integration constants known, the displacements are also known. It should thus be evident that, once you have formulated the boundary conditions as mathematical expressions, you can solve for the reaction forces and deflections of any statically indeterminate beam - regardless of the degree of indeterminacy: one could add another boundary condition, and this would simply result in another unknown (another reaction force) and another equation (from the boundary condition), such that the system remains of full rank. Thus, although the equations are sometimes relatively lengthy (and the resulting matrix is moderately dense), this approach of using Macaulay step functions and integrating the moment curvature relations is guaranteed to provide a solution, and it is merely a matter of correctly programming the method.

Suggested simplifications

Although the method outlined above is simple and robust, there are a couple of characteristics that make it challenging to implement. These are:

- The aileron is rotated an angle θ before the loads act on it. This does not pose any real issues for the boundary conditions at the hinges (the vertical displacements there can be straightforwardly decomposed along the rotated coordinate system), but does make the boundary condition at the stuck actuator harder to implement, as there is now a boundary condition on a linear combination of the deflections due to bending and twist (since the actuator does not act through the shear center). Conceptually, it is not hard at all to represent this boundary condition, but the implementation is very prone to minus sign mistakes and other coding mistakes, simply due to the large number of terms that will appear in this equation.
- The aerodynamic load is two-dimensional and discrete. You would first need to interpolate the data set, and then integrate the load up to five times (once in z-direction, four times in x-direction). Although there are `numpy` and `scipy`-functions to aid you in this, this may still take some time.

Therefore, you are suggested to make the following simplifications in your model:

- When setting up the boundary conditions for the deflections, you are suggested ignore the contributions due to twist, in other words, in Equation (17)-(19), do not include the presence of the term involving $\theta(\cdot)$. This removes one of the unknowns from the system of equations (namely C_5, the integration constant in the twist), so there is a need to alter the boundary conditions slightly as well.
  Therefore, you are suggested to remove the boundary condition at the jammed actuator (do not remove the reaction force that is present there, though). This ensures you have a closed system again[3].
  You can solve for C_5 by adding a fictitious boundary condition directly on the twist somewhere along the aileron (e.g. at one of the actuators or at one of the hinges), setting the twist at that position equal to a value you find suitable (e.g. 0).
  Mathematically speaking, choose a position x_c and a value of the twist ϕ, and add the boundary condition $\theta\left(x_c\right) = \phi$ (*without* introducing another reaction torque at this location, which would normally be there). It is up to you to choose a position x_c and a a suitable value ϕ.[4].
- Do not implement the two-dimensional aerodynamic load in your own numerical model, but simply ignore the two-dimensional aerodynamic load and instead implement a one-dimensional, uniformly distributed load along e.g. the hinge-line or the quarter-chord. This makes integration of the aerodynamic load quite trivial.

You can be assured that these simplifications end up with results that still are sufficiently accurate to be useful in verification of the developer model (assuming that the rest of your numerical model works correctly). Nonetheless, it is still up to you to justify these simplifications, and, as stated in the assignment, argue why your numerical model is still sufficiently accurate.

In short, these simplifications should help you strike a right balance between accuracy of the numerical model and having time to focus on the actual verification of the developer model. After all, as mentioned in the assignment, your verification of the developer model should *also* rely on independent tests of the developer model; some of those tests will likely also take some time to develop and implement so you should divide your resources appropriately.

[3]In theory, you could remove *any* of the boundary conditions and keep the boundary condition at the actuator instead. Try to think about why removing the boundary condition at the actuator is the preferred boundary condition to remove.

[4]Note that there are suitable positions and values that you can choose for this; there is not really a single correct ‘solution’ for this.