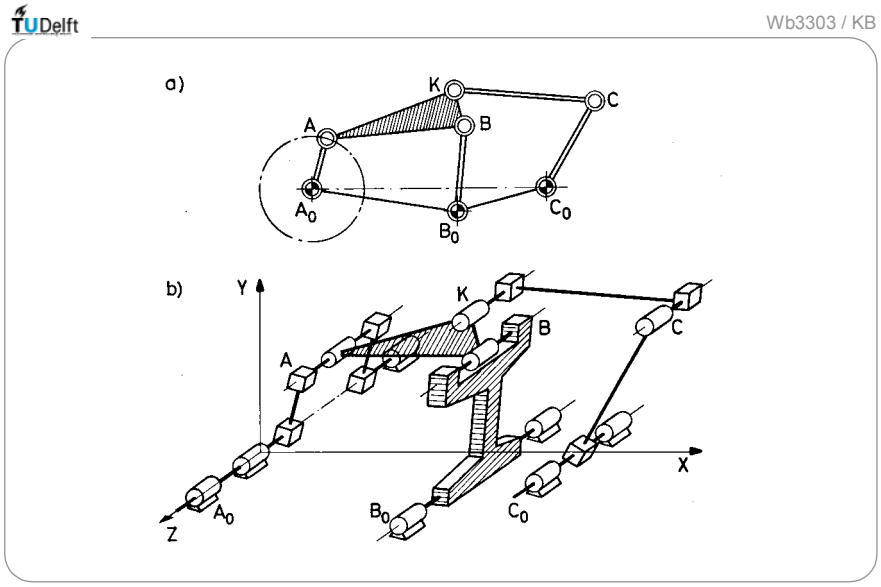

Fig. 8.1.1 Embodiment properties in a co-planar mechanism model

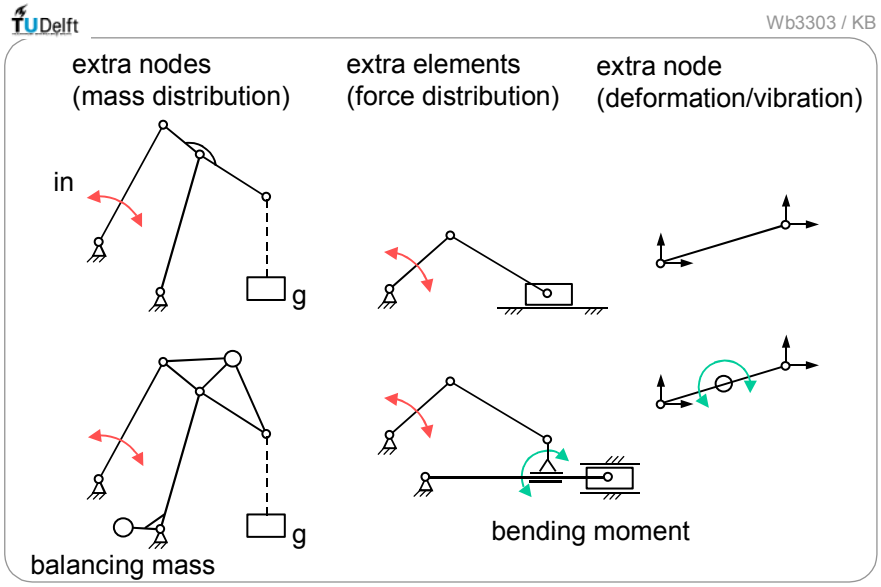

Fig. 8.1.2 Typical adaptation of topology for dynamics

8 kineto-statics and Dynamics

8.1 Introduction

Dynamics = branch of theoretical mechanics dealing with the motion and equilibrium of bodies and mechanical systems under the action of forces (IFToMM [1.1]).

Kineto-statics: this term will be used here considering the mechanical system as a rigid body system with given time behaviour of its motion.

Dynamics is the more general term. Kineto-statics deals with a limited field of dynamics: the theory that can be used as if the mechanical system is, in a certain position, a static construction. Timed motion, and thereby acceleration, plays anyhow a role. The forces include certainly the mass forces, both in kineto-statics and dynamics.

The intention of this chapter should be regarded in the context of the design process of mechanisms, see fig. 1.3.1. It will be assumed that the kinematics part of the mechanism design has been completed: the mechanism type has been chosen and the kinematic dimensions are given quantities. The forces acting at the individual parts are to be calculated now. The results are needed for the design of bearings, the choice of a driving motor, the thickness dimensions of the links and so on.
Typically the dynamics part has iterative loops. When thickness dimensions and the construction material have been chosen, the masses of the parts are known. But the masses are needed to calculate the mass forces, so usually the stresses and deformations can only be verified properly after having made this choice. A lot of measures deserve attention for better dynamic behaviour, like adding extra mass or springs (balancing of forces), shape of the parts and so on.
Finally the mechanical model should be regarded as a three-dimensional model, even when it is a co-planar mechanism (fig. 8.1.1). During the various design stages it will frequently be needed to adapt the model (with respect to its physical properties) to the design purpose. Some examples of adaptation are shown in fig. 8.1.2. The dynamic model will occasionally be much more extended compared with a pure kinematic model, but the kinematic motion will be the same.

It will be clear that the Finite Element approach is very suitable to handle such model definitions and extensions towards dynamics and kineto-statics. The examples show that many of the design decisions in dynamics don't need a three-dimensional mechanism model (for a co-planar mechanism). The more complex 3D-model can thus be avoided during a great part of the conceptual design of the mechanism. This book concentrates therefor on the theory that can be used with a 2D-model of a planar mechanism, which is kinematically determined (the matrix D^{cp} is regular).

It will be assumed further:

- The parts will have only small deformations due to the forces. It means that the parts are fairly rigid, but it is the intention to find out which flexibility is allowed. The principle of superposition can be used:

  The total motion of the mechanism (co-ordinates) is the sum of rigid motion and displacement due to elasticity of the links.

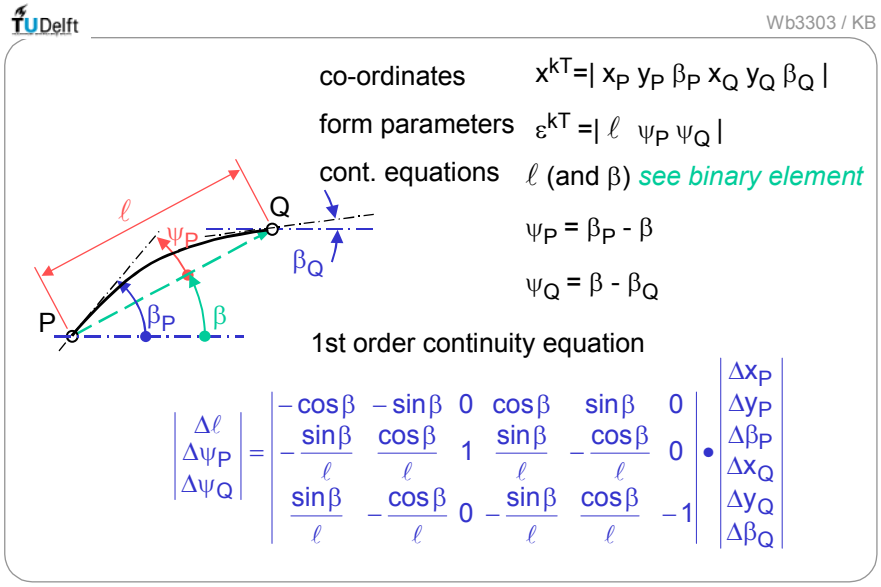

Fig. 8.1.3 Beam element, special for dynamics

- Friction will, in the conceptual design phase, mainly be important for mechanisms that are supposed to jam. In the majority of cases it will be tried to keep the friction low, allowing to neglect friction in the mechanical system. It is beyond the scope of this book to include friction in the theory.

To be better prepared for dynamics a new element will be introduced here: the planar beam element (see fig. 8.1.3). It has the characteristic property that, compared with the binary element, bending deformation has been added. Naturally the two form parameters describing the bending (ψ_P and ψ_Q) should be understood as prescribed form parameters. The major advantage of this element is that the internal force distribution (and deformation modes) can, in specific cases where bending in a bar is significant, better be modelled. In a later chapter (8.6.2) it will be shown that the two bending deformations are coupled by the bending stiffness of the beam. This property cannot be combined with the idea that a deformation should be understood as a large deformation, as required to model input motion. When a beam is to be used as a driven crank, one of the angular co-ordinates would be a proper quantity to use as the crank angle. In this book however only form parameters are used as driving quantities. Adding a half beam element (extra element) can solve this problem, sharing the angular co-ordinate with the beam. The half beam has the proper form parameter to drive the mechanism.

8.2 Virtual work and equilibrium equations (statics)

8.2.1 General principle

The mechanism can be considered as a flexible construction with the input motion " frozen" (rigid, but flexible). At such a construction forces can be applied, just like in statics. In this conceptual mechanism model the driving motion quantities (concerning the degrees of freedom) will be treated like all other prescribed form parameters.
In statics the principle of virtual work is the same as the condition for equilibrium:

> *If the virtual work of the applied (external) forces is equal to the virtual work of the internal forces of the mechanism, then there is equilibrium.*

The reverse statement is also true: If there is equilibrium in the mechanical system, then the virtual work of applied forces and internal forces are equal.
The virtual work of the applied forces must be considered with the displacements of the co-ordinates (of the nodes and other types of co-ordinates), as they can be moved kinematically. This implies, for instance, that a fixed point ($\delta x^o = 0$) cannot contribute to the amount of virtual work:

$$\delta W_{ex} = \langle f , \delta x \rangle = \langle f^c , \delta x^c \rangle \tag{8.1}$$

The index c indicates the moving co-ordinates of the mechanism. The <> brackets indicate the inner (scalar) product of two vectors from different physical vector spaces.
Co-ordinates are kinematically movable due to allowed deformations (caused by the forces) of the prescribed form parameters. Actually this is a choice that can be made for each mechanism model. Suppose for instance that all prescribed form parameters are considered as undeformable (really rigid), then the amount of virtual work of the applied forces can only be zero. In general however one or more prescribed form parameters will be considered as deformable, allowing them to absorb the virtual work. Assume now that one particular form parameter ε^p is deformable:

$$\delta W_{in} = \langle \sigma , \delta \varepsilon \rangle = \langle \sigma^p , \delta \varepsilon^p \rangle \tag{8.2}$$

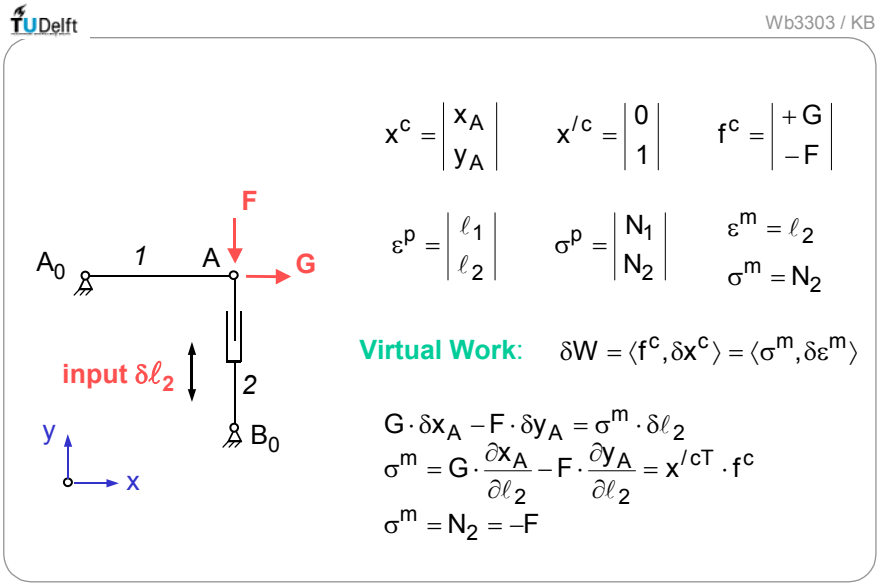

Fig. 8.2.1 *Driving force σ^m=N_2 as an internal force (normal force) Example 1*

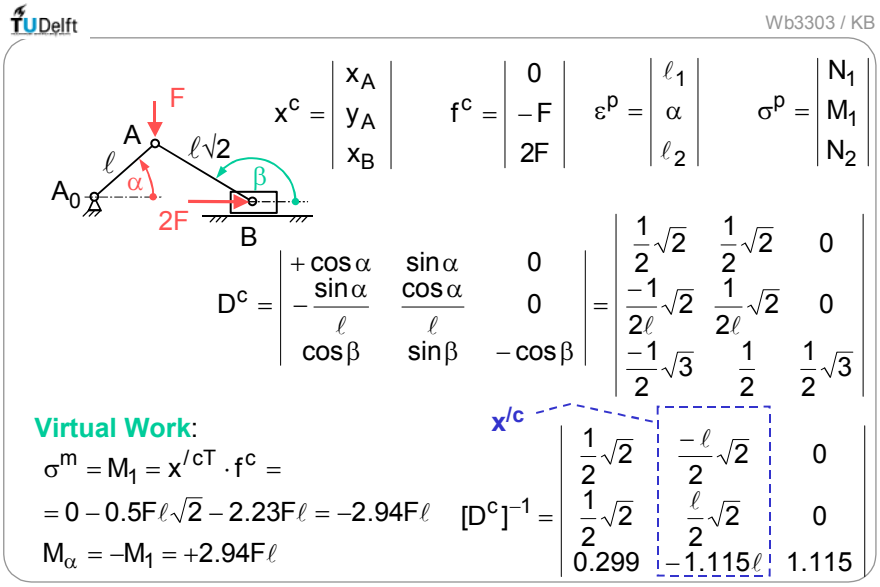

Fig. 8.2.2 *Driving moment σ^m =M_1 as a generalized internal force (example 2)*

The coefficient of vector σ^p contains the (internal) force that, when multiplied with the corresponding deformation, yields the virtual work. In the equilibrium situation it must hold that:

$$\langle f^c , \delta x^c \rangle = \langle \sigma^p , \delta \varepsilon^p \rangle \tag{8.3}$$

How the co-ordinates can move kinematically is expressed by the first order transfer function (5.16), noted here as:

$$\delta x^c = \frac{\partial x^c}{\partial \varepsilon^p} \cdot \delta \varepsilon^p \tag{8.4}$$

After substituting (8.4) in (8.3) the following vector equation can be obtained:

$$f^{cT} \frac{\partial x^c}{\partial \varepsilon^p} \delta \varepsilon^p = \sigma^{pT} \delta \varepsilon^p$$

Dividing at both sides by $\delta\varepsilon^p$, which is by definition non-zero, it follows that

$$\sigma^{pT} = f^{cT} \frac{\partial x^c}{\partial \varepsilon^p} = \left[\frac{\partial x^c}{\partial \varepsilon^p} \right]^T \cdot f^c \tag{8.5}$$

Assuming that all prescribed form parameters $\delta\varepsilon^p$ are independent variables, the result (8.5) can be written in one equation system for all possible deformations $\delta\varepsilon^p$:

$$\sigma^p = \left[\frac{\partial x^c}{\partial \varepsilon^p} \right]^T \cdot f^c = \left[D^{pcT} \right]^{-1} \cdot f^c \tag{8.6}$$

It can be concluded: the internal forces σ^p and the applied (external) forces f^c are related by the first order kinematic transfer functions.

Note that the description of forces, both the internal and the external forces, is to be understood in the FEM in the generalized way. It depends on the modelled definition of the corresponding co-ordinate (or form parameter respectively) which physical dimension the force has. The multiplication of

- external force and co-ordinate displacement, or
- internal force and deformation

must yield (virtual) work.

8.2.2 Driving forces

A specific application of (8.5) concerns the driving forces of the mechanism. In the FEM-approach they are considered as internal forces (one or more coefficients of the σ^p –vector). Writing these driving forces explicitly:

$$\sigma^m = x'^{c\,T} \cdot f^c \tag{8.7}$$

it can be recognized easily that the applied forces must be multiplied with the vector of first order kinematic transfer functions to obtain the (static) driving force to make equilibrium with the external forces.

The example of figure 8.2.1 concerns a simple mechanism with two links, in a situation that they are perpendicular. The vertical link is a driving cylinder (form parameter δP_2). There is one moving node A at which forces +G and -F exert:

$$f^c = | +G \;\; -F |^T$$

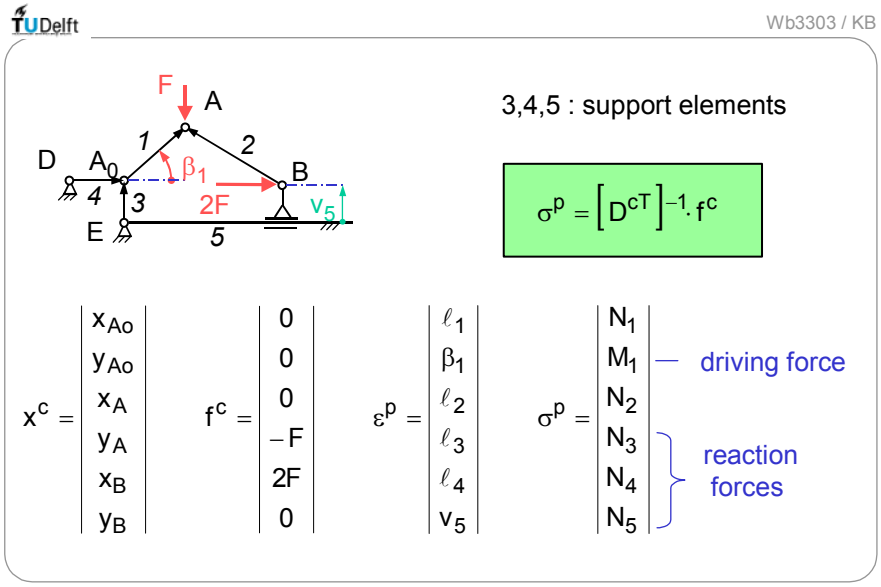

Fig. 8.2.3 Reaction forces, implicit method

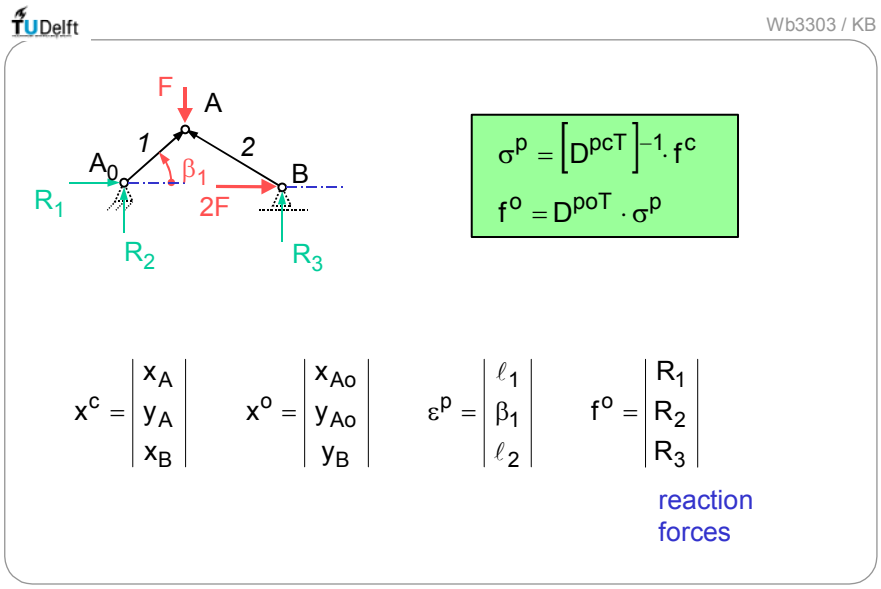

Fig. 8.2.4 Reaction forces, explicit method

In this special position the first order transfer function can be found intuitively as:

$$x^{/cT} = |\ 0\ \ 1\ |$$

and the vector product (8.2) yields that the driving force (normal force N_2) equals –F (pressure).
The example of figure 8.2.2 shows a slider-crank mechanism with external force F at node A and 2F at node B. To find the vector $x^{/cT}$ the matrix D^{pc} can be written out and filled with its numerical values (using here α=45° and β=120°). The result of manual matrix inversion is presented in the figure. Because the driving quantity α is the second form parameter (second row of matrix D^{pc}), the vector of first order transfer functions $x^{/c}$ is the second column in the inverse of matrix D^{pc}. Applying (8.7) the driving force, to be understood here as an internal driving moment $\sigma^m=M_1$, can be calculated (M_1 = -2.94F ℓ). The minus sign indicates that the internal force is to be understood as a compressive force, since it tries to decrease the corresponding form parameter (driving angle, compare the cylinder in the previous example).

8.2.3 Support forces

Two methods to find the support forces (reaction forces exerting at the frame) will be considered.

Implicit calculation.
The model can be extended with extra elements, such that the fixed nodes will be connected to the frame by prescribed form parameters (conceptually the same trick as in kinematic optimization). The same mechanism as in the previous chapter will be used as an example, see fig. 8.2.3. Point A_0 has been connected by binary links (elements 3 and 4) to the frame. The normal forces in these bars (N_3 and N_4) will be coefficients of the vector of internal forces σ^p, when applying the general rule (8.6). These normal forces are then the support forces of node A (but mind the direction: a positive value means a tensile force, a negative value means a compressive force).

Explicit calculation.
The reaction forces need to be included directly in the method. Now the displacement of fixed co-ordinates δx^o must be understood as nonzero, otherwise the reaction forces cannot contribute to virtual work. It will be sufficient to consider all form parameters as undeformable ($\delta\varepsilon^p$=0). Equilibrium of the external forces requires then that

$$\delta W_{ex} = \langle f\ ,\ \delta x \rangle = \langle f^c\ ,\ \delta x^c \rangle + \langle f^o\ ,\ \delta x^o \rangle = 0$$

Referring to eq.(5.14) it follows that

$$\delta\varepsilon^p = D^{pc} \cdot \delta x^c + D^{po} \cdot \delta x^o = 0 \quad \text{and}$$

$$\delta x^c = [D^{pc}]^{-1} \cdot D^{po} \cdot \delta x^o$$

must be applied instead of eq. (8.4). The result becomes:

$$f^o = \left[D^{poT}\right] \cdot \left[D^{pcT}\right]^{-1} \cdot f^c = \left[D^{po}\right]^T \cdot \sigma^p \qquad (8.8)$$

In other words: the supporting forces can be calculated directly after that the internal forces are known. A specific part of the D-matrix, referring to the fixed co-ordinates and prescribed form parameters $[D^{po}]$, is needed here.
In figure 8.2.4 the model of the sample mechanism, with which the reaction forces can be found explicitly, has been depicted.
In the present version of RUNMEC the explicit method is not supported.

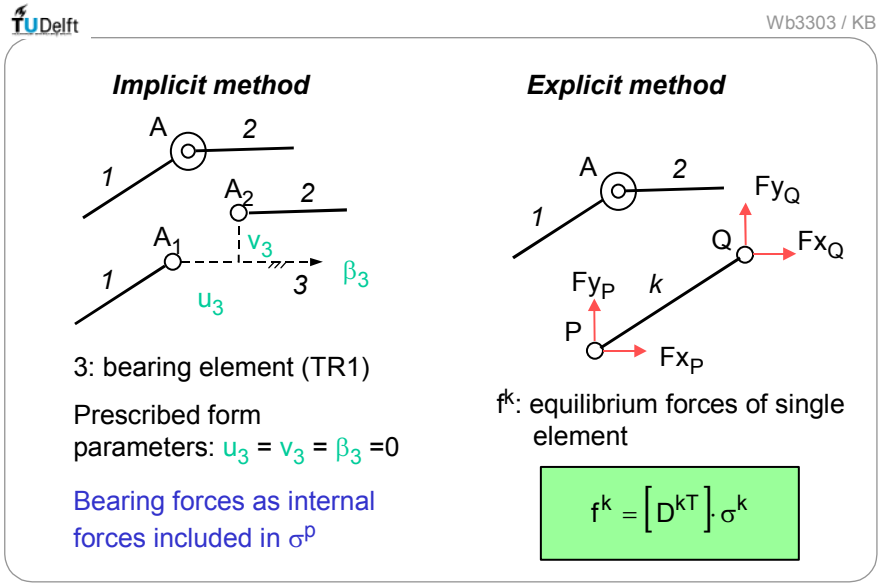

Fig. 8.2.5 Connection forces (bearing forces) in a hinge

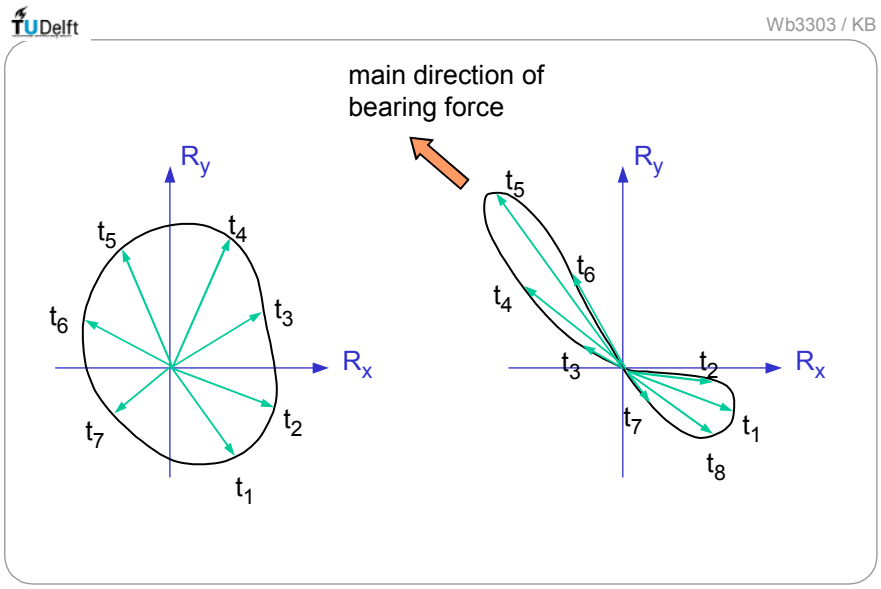

Fig. 8.2.6 Typical behaviour of bearing force (polar diagram)

8.2.4 Connection forces

Comparable with the support forces the connection forces (bearing forces) can be determined by two different approaches.

Implicit method.
An extra element is used to model the bearing forces as internal forces. For a revolute joint the ternary element-1 is typically suited for this task, see fig. 8.2.5 (left part). The two neighbouring elements must not share a node, but get a node each (A_1 and A_2) that will be shared with the connecting element. The two points A_1 and A_2 can coincide by prescribing the form parameters of the connecting element (u = v = 0). The angular co-ordinate of the ternary element is to be understood as fixed and will usually chosen to be zero.
An advantage of this method is that the mass of the bearing house can properly be divided over the two nodes involved with the connection. Disadvantage is that the model (the dimension of the D^c-matrix) becomes bigger.

Explicit method.
Assume that the internal forces have already been calculated (the σ^p-vector is known). One of the elements having the connection is to be "disassembled", see fig. 8.2.5 (right part). The connection forces of this element are to be understood now as (part of) the total external forces (equilibrium forces f^k_{eq}) of that element number k. Naturally the equilibrium forces must cause the known internal forces σ^k of that element:

$$\langle f^k_{eq} , \delta x^k \rangle = \langle \sigma^k , \delta \varepsilon^k \rangle \tag{8.9}$$

Applying the continuity equation $\delta\varepsilon^k = D^k \cdot \delta x^k$ it follows that

$$f^k_{eq} = D^{kT} \cdot \sigma^k \tag{8.10}$$

The conclusion is that the continuity matrix D^k of just that element is required to calculate the equilibrium forces of that element. In these values the connection forces of the bearing under consideration are comprised.
The explicit method looks simple enough to perform occasionally a manual calculation or verification. It must be noticed however that the external forces f^k, exerting at the element co-ordinates (including mass forces, see next chapter), contribute to the equilibrium forces. The connection forces (f^k_{con}) are then:

$$f^k_{con} = f^k_{eq} - f^k \tag{8.11}$$

An ambiguity arises when the mass distribution of the bearing house comes into account. (Mass) forces can only be combined with displacements of co-ordinates. Using the shared node connection of the two links, a further specification is needed which part of the bearing house mass belongs to the disassembled element. This specification needs to be introduced here manually in the vector f^k.

In many design cases the final shape of the elements will be dominated highly by the construction of the bearings. The calculation of the connection forces, usually for worst case assumption of external forces, is certainly required for the bearing design (or to select bearings from a catalogue). Fig. 8.2.6 shows some examples of typical bearing force behaviour in a polar diagram. Such information is a basis for the primary decisions about the bearing design.

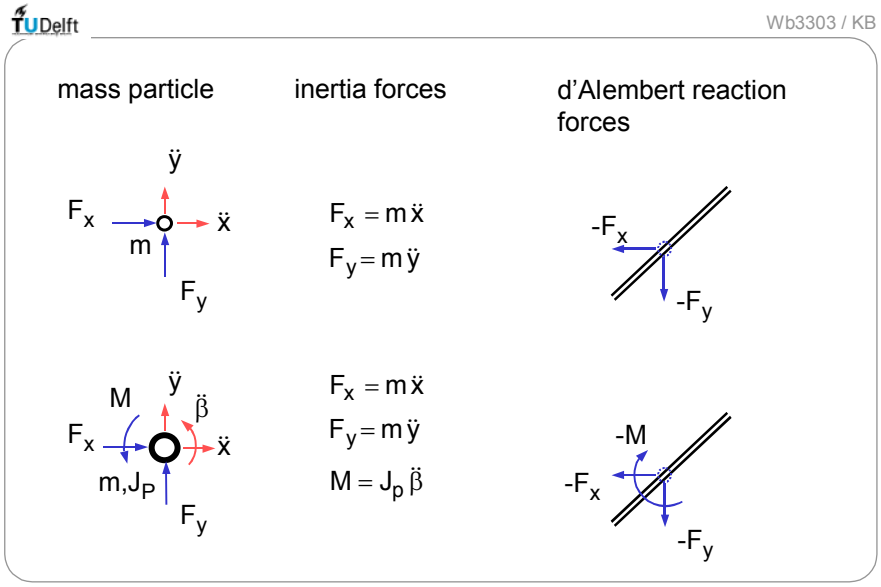

Fig. 8.3.1 *Replacement of mass by external forces (d'Alembert principle)*

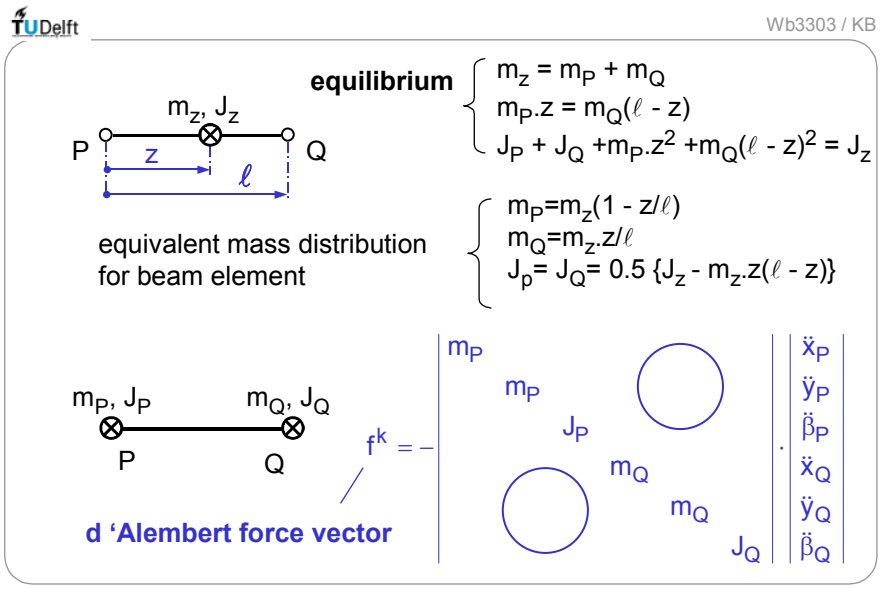

Fig. 8.3.2 *Equivalent mass distribution for a beam element, without using a centre of mass*

8.3 Mass forces

8.3.1 Principle of d'Alembert

According to Newton the force components to give a particle (a point mass m) acceleration are:

$$F_x = m\ddot{x}$$

$$F_y = m\ddot{y}$$

These force components must be delivered by the element the particle belongs to. The particle exerts thus a reaction force at the element (d'Alembert principle, see fig. 8.3.1):

$$\begin{aligned} F_x &= -m\ddot{x} \\ F_y &= -m\ddot{y} \end{aligned} \tag{8.12}$$

These forces will be considered thus as external forces, which replace the mass.
In many cases a wheel-like part can be modelled as a point mass with finite rotational (polar) inertia J_p. The corresponding reaction moment according to d'Alembert is then:

$$M_\beta = -J_P\ddot{\beta} \tag{8.13}$$

In the FEM-approach external forces must be related to a (moving) co-ordinate. The d'Alembert forces have only meaning when the corresponding co-ordinate is available in the mechanism model.

8.3.2 Mass centre and beam element

In many cases an element with length (binary element, beam) has, more or less, a uniform mass distribution. To describe a mass distribution in the FEM approach a proper set of element co-ordinates is required to connect the d'Alembert forces. A distributed mass needs thus to be replaced by some discrete mass distribution. Two concepts are useful, see fig. 8.3.2:

With a mass centre.
An extra node for the mass centre will be used. Now the mass m_z of the bar can be concentrated as a point mass in the mass centre. Usually the bar has also a polar inertia, so the model needs also a rotational co-ordinate at the extra node. It looks natural to use two beam elements rigidly connected in the mass centre. There are now only mass forces acting at the mass centre point, which are equivalent (with respect to the equilibrium of forces) with the real mass forces.

Without a mass centre.
No extra point will be introduced, so the equivalent mass forces can only act at the existing co-ordinates of the bar or beam. The total mass m_z could be divided over the two end-points P and Q of the bar, proportional to their distance of the (virtual) mass centre (see fig. 8.3.2 lower part). But this affects also the polar inertia of the link, which will become now:

$$m_P \cdot z^2 + m_Q(\ell - z)^2$$

The really existing polar inertia J_z ($= \frac{1}{12} m\ell^2$ for a uniform distribution) can be corrected however by compensating rotational inertia at the end points, which must have then a rotational co-ordinate. The beam element is very well suited for this task: it has a rotational co-ordinate at both end points. A choice can be made now where to compensate: either at P or at Q (or half at both points, as suggested in fig. 8.3.2).

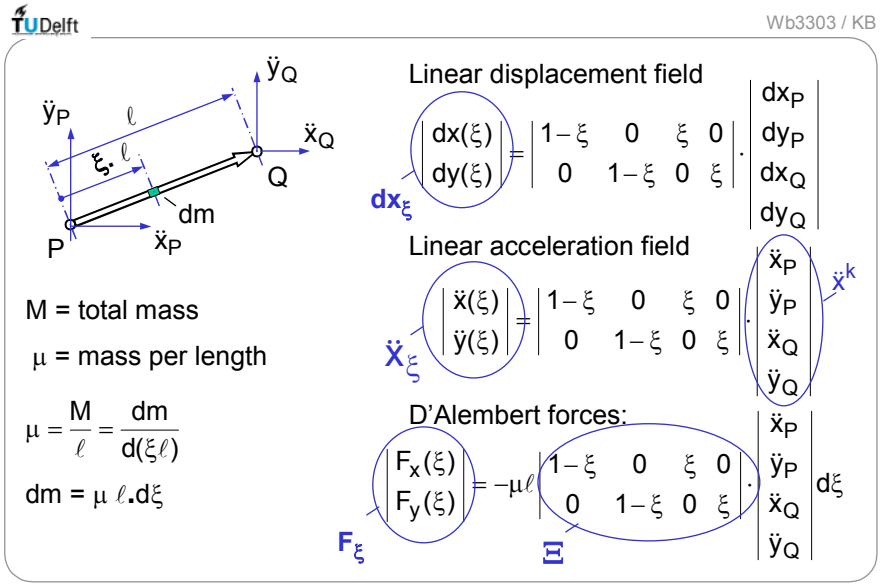

Fig. 8.3.3 *Uniformly distributed mass of a binary element, contribution of a particle dm*

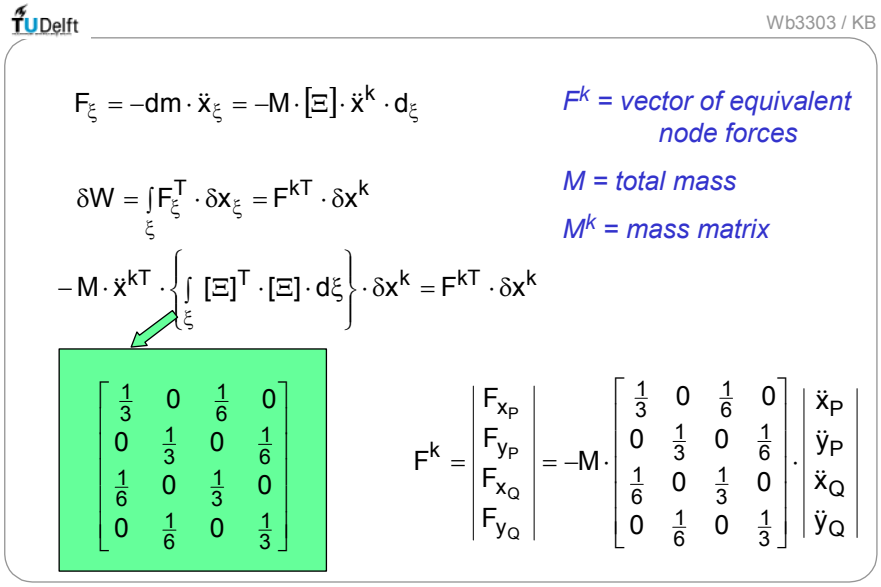

Fig. 8.3.4 *Mass matrix of a binary element (uniformly distributed mass), using equivalent node forces*

Both concepts specify a solution for equivalent mass forces, which means they are correct with respect to the equilibrium of the forces. The (internal) force distribution is possibly only an approximation (discussion later in chapter 8.3.5). Both concepts couple directly d'Alembert forces to co-ordinates. Therefor the (vector of) mass forces can be expressed simply using a diagonal mass matrix, see fig. 8.3.2.

8.3.3 Mass matrix of the binary element

The bar PQ, see fig. 8.3.3 has a uniform mass distribution. The total mass is M and the specific mass (mass per length unit) is denoted μ. The intention is to find the equivalent mass forces (vector F^k, its components acting at the four co-ordinates of this element).
The bar could be divided into many particles with mass dm at line PQ. A particle at position ξ along the bar will be displaced dependent on the displacements of the nodes P and Q. Since the displacement field is linear, a linear matrix Ξ can be constructed to express the displacement (dx, dy) of the mass particle:

$$dx_\xi = \begin{vmatrix} dx(\xi) \\ dy(\xi) \end{vmatrix} = \begin{vmatrix} 1-\xi & 0 & \xi & 0 \\ 0 & 1-\xi & 0 & \xi \end{vmatrix} \cdot \begin{vmatrix} dx_P \\ dy_P \\ dx_Q \\ dy_Q \end{vmatrix} \tag{8.14}$$

Observing that the acceleration field is also linear (see chapter 3.3.2) the same matrix Ξ can be used to express the acceleration of the particle, dependent on the acceleration of P and Q (vector $\ddot{x}^k$). The d'Alembert forces at the particle

$$F_\xi = -dm \cdot \ddot{x}_\xi = -M \cdot d\xi \cdot [\Xi] \cdot \ddot{x}^k \tag{8.15}$$

must give the same amount of virtual work as the equivalent node forces F^k:

$$\delta W = \int_\xi F_\xi^T \cdot \delta x_\xi = F^{kT} \cdot \delta x^k$$

Substitution of (8.14) and (8.15) leads to (see also fig. 8.3.4)

$$-M \cdot \ddot{x}^{kT} \cdot \left\{ \int_\xi [\Xi]^T \cdot [\Xi] \cdot d\xi \right\} \cdot \delta x^k = F^{kT} \cdot \delta x^k$$

in which the integral at the interval $0 \le \xi \le 1$ yields the matrix

$$\begin{bmatrix} \frac{1}{3} & 0 & \frac{1}{6} & 0 \\ 0 & \frac{1}{3} & 0 & \frac{1}{6} \\ \frac{1}{6} & 0 & \frac{1}{3} & 0 \\ 0 & \frac{1}{6} & 0 & \frac{1}{3} \end{bmatrix}$$

This matrix, multiplied with the total mass M of the element, is known as the mass matrix M^k. The final result, the equivalent mass forces F^k, can be expressed then as:

$$F^k = \begin{vmatrix} F_{x_P} \\ F_{y_P} \\ F_{x_Q} \\ F_{y_Q} \end{vmatrix} = -M \cdot \begin{bmatrix} \frac{1}{3} & 0 & \frac{1}{6} & 0 \\ 0 & \frac{1}{3} & 0 & \frac{1}{6} \\ \frac{1}{6} & 0 & \frac{1}{3} & 0 \\ 0 & \frac{1}{6} & 0 & \frac{1}{3} \end{bmatrix} \cdot \begin{vmatrix} \ddot{x}_P \\ \ddot{y}_P \\ \ddot{x}_Q \\ \ddot{y}_Q \end{vmatrix} \tag{8.16}$$

Note that this mass matrix has also terms off the diagonal. Considering the internal forces, this mass model will not describe the bending effect due to the uniform mass.

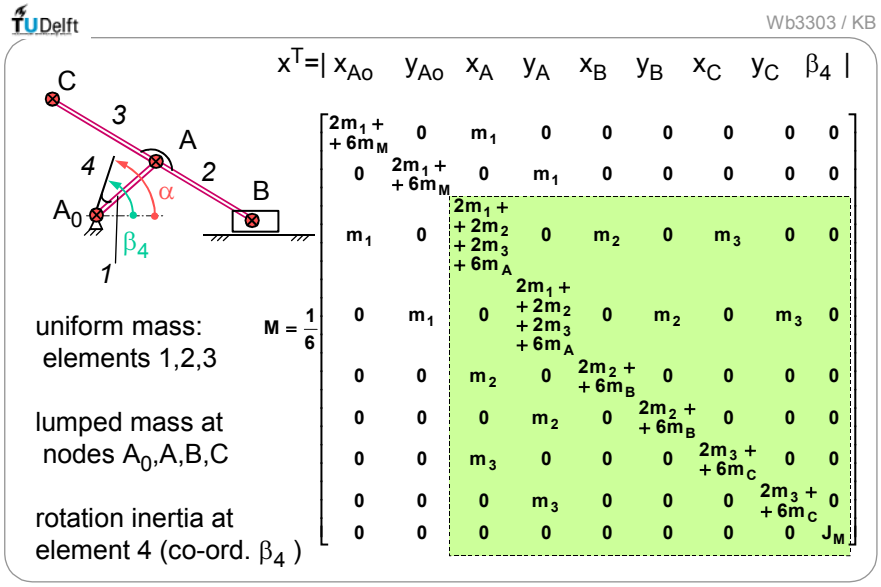

Fig. 8.3.5 Mass matrix of a mechanism (example)

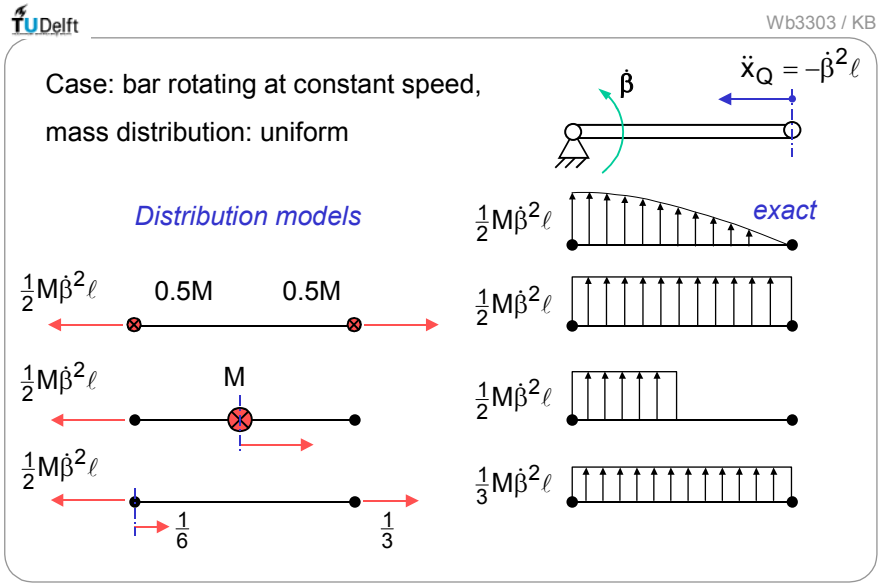

Fig. 8.3.6 Internal force distribution (example)

8.3.4 Mass distribution of a mechanism (example)

The mechanism of fig. 8.3.5 will be considered as an example. The three links have a uniform mass distribution. In addition the points A_0, A, B and C have been given point masses (to indicate the mass of the bearing houses). The polar inertia J_M at the input shaft (driving motor, gear-box, coupling, fly-wheel) needs an angular co-ordinate, so the half beam element 4 has been connected to the crank A_0A. Now the assembled mass matrix of the whole mechanism can be written out, see the figure 8.3.5. The contributions of the mass specification, referring to the same matrix coefficient, can simply be added.

8.3.5 Internal force distribution

Mass forces can only be connected to co-ordinates of the mechanism model. Especially when a uniform mass distribution of an element comes into account, the effect to the internal force distribution should be considered.
The example of figure 8.3.6 shows a mechanical system consisting of one bar rotating with constant speed $\dot{\beta}$ around one of the end-points (rotating crank). The normal force in the bar will be considered. The exact internal force distribution is some (quadratic) function of the radius: the maximum is $\frac{1}{2}M\dot{\beta}^2$ at the centre, the minimum 0 at the end.

- In a mass distribution with the mass divided equally over the two end-points, the normal force would be constant at $\frac{1}{2}M\dot{\beta}^2$.

- The mass concentrated in the mass centre would give a constant normal force of $\frac{1}{2}M\dot{\beta}^2$ in the inner half of the beam, and zero force in the outer half of the beam.

- A mass distribution according to the mass matrix of eq. (8.16) would specify a constant normal force at $\frac{1}{3}M\dot{\beta}^2$.

It will be clear that none of these mass distributions is correct. It depends on the user's choice which mass distribution will be preferred.
In a conceptual phase of the design however all models are qualitatively correct regarding the maximum value of the force.

A comparable note can be made with respect to the deformations of the element, as caused by the forces. They will certainly not be exact, but might be useful as an indication in the conceptual design phase (more about deformations in chapter 8.6).

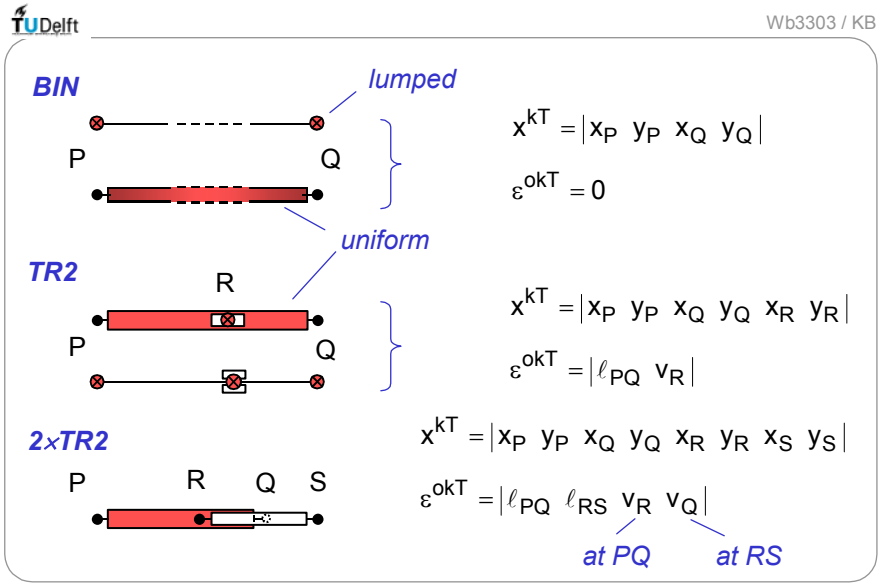

Fig.8.3.7 Some mass distribution options with a slider pair

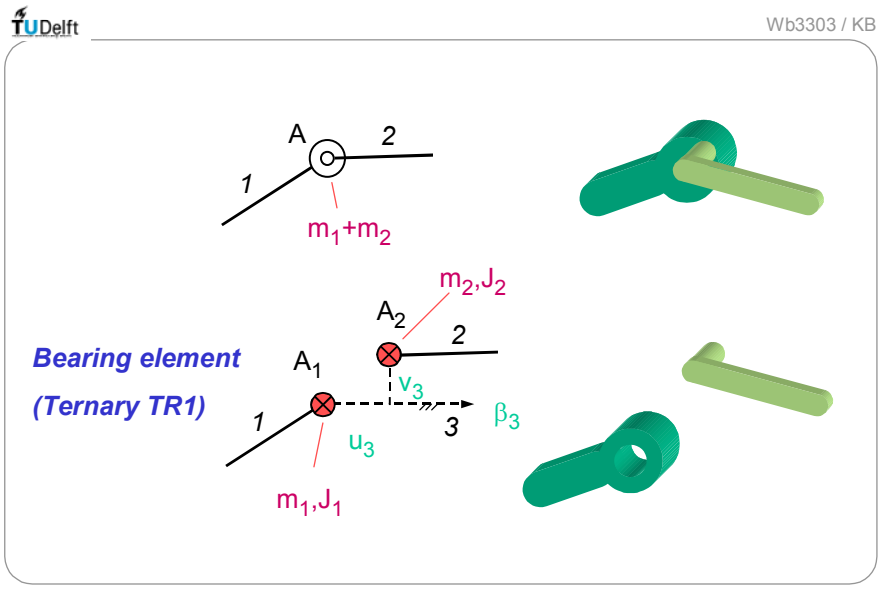

Fig. 8.3.8 Mass distribution of a revolute joint (example)

8.3.6 Mass distribution of joints

A *sliding joint* can in the FEM-approach easily be modelled with a binary element, by assuming the length non-prescribed. Suppose now that a uniform mass distribution will be taken. In that case the specific mass (mass per length unit) would vary and this is physically not realistic.

Sliding joint models, which are more realistic with respect to the construction, are depicted in fig. 8.3.7.
In many cases one of the sliding elements (the guide) is beam-like and can be given a uniform mass, see the middle part of the figure. The other sliding element is short and can be regarded as a point mass. The ternary element(2) is very well suitable to specify such a mass distribution. Using this element the internal force referring to form parameter v is the bearing force (perpendicular to the guiding element).
When the slider pair consists of two telescopic cylinders, a uniform mass distribution could be proposed using two ternary(2) elements. Mutually they share a third point and an end-point, as indicated in the figure (lower part). Notice that the distance QR is not allowed to be zero (but in the real construction this is also not allowed). The two internal forces, referring to the form parameters v_Q and v_R, are the bearing forces.

A *revolute joint* deserves attention with respect to the mass of the bearing house. In addition to the way a (static) bearing force can be modelled implicitly using a help element (preferably a ternary-1 element, see fig. 8.2.5), the mass of the bearing house can be split up as indicated in fig. 8.3.8. When polar inertia is significant, an angular co-ordinate must be present at the node. In that case a ternary element(1) or a beam element could be used.
Occasionally the ideas expressed in the figures 8.3.7 and 8.3.8 need to be combined. For instance: when the revolute joint specification is required at the sliding point it can be necessary then to add extra (dummy) elements, in order to create an angular co-ordinate.

8.4 Other external forces

8.4.1 Springs

Characteristic for a spring is, that the (kinematic) degree of freedom of the mechanism remains unchanged when a spring has been added. By its nature a spring can have an internal force and a large deformation. In the FEM-approach these characteristics of the spring can be maintained while the spring forces are converted to external (reaction) forces. Since such a force is dependent on the geometry of the mechanism, ***it needs to be coupled to a dependent form parameter***.

A typical spring has been depicted in fig. 8.4.1. This (coil) spring can be represented by a binary element, but with non-prescribed length. The idea is, that the relation between the internal force σ (here the normal force F_N) and the form parameter (here the length of the link), usually indicated with the ***spring characteristic***, is given. The spring element will be disassembled to find the equilibrium forces f^k using eq.(8.10). The reaction forces $-f^k$ are then the (external) forces exerted by the spring at the co-ordinates of the mechanism.
This idea is general for all kinds of dependent form parameters. All kinds of springs can thus be modelled. Example: a flat spiral can be coupled to the angle between two bars (the difference angle needs to be composed by subtraction of the two angular form parameters).
Although in most cases springs will be assumed to have no mass, all mass specifications concerning the spring element can normally be used.

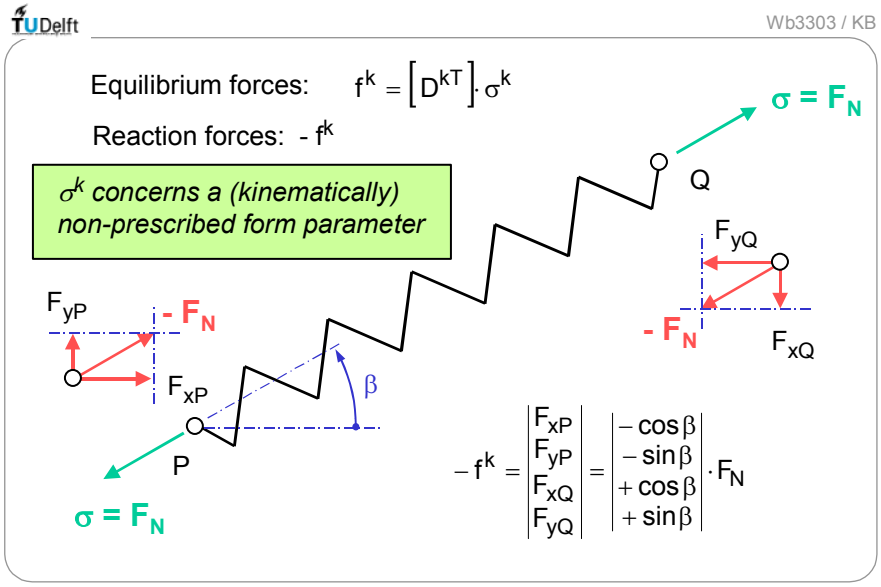

Fig. 8.4.1 *Spring force as an external (reaction) force at connecting nodes*

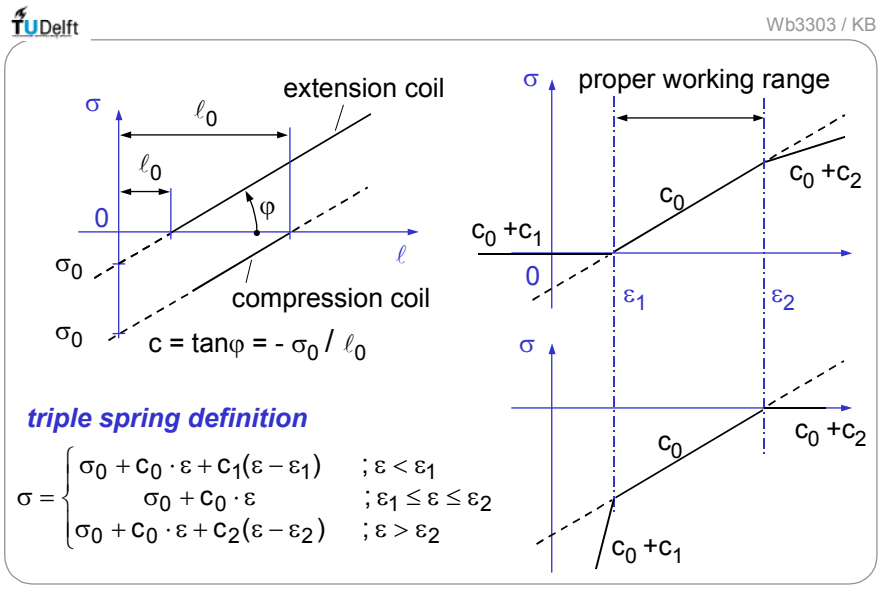

Fig. 8.4.2 *Linear spring and working range (generalized definition)*

Usually a linear spring characteristic comes into account, like depicted in fig. 8.4.2 (upper left). The spring characteristic is then typically, with c as the spring constant (stiffness):

$$\sigma(\varepsilon) = c \cdot (\varepsilon - \varepsilon_0) \tag{8.17}$$

For the coil, coupled to a binary element, this spring characteristic can be interpreted as:

$$\sigma(\ell) = c \cdot (\ell - \ell_0) \tag{8.18}$$

Notice the typical difference between an extension coil and a compression coil with respect to the working rang of the internal force.
It looks a good idea to define the (linear) spring characteristic in a special way: by the value of c (spring constant) and the value of $\sigma(0)$. The advantage of this method is that it is also possible now to specify a ***constant force with variable direction***. The spring element is borrowed for this purpose, specifying c = 0 and $\sigma(0)$ as the constant force.

Realistic springs usually have a working range. A compression coil for instance has in its released state (and for greater length) a zero force. In the state of minimum length all windings touch and the stiffness increases enormously. An extension coil will be slack below a certain minimum length and be permanently deformed beyond its maximum allowable length.
A triple range definition of a (linear) spring, see figure 8.4.2 (right and lower part), may define more realistic spring forces. The proper working range, the middle part of the spring characteristic having stiffness c_0 between the values ε_1 and ε_2, can be defined as above. The state outside this range needs a correction of the stiffness, as proposed here with values c_1 and c_2 . In the Runmec program this triple spring characteristic has been implemented.

8.4.2 Damping forces

Viscous damping (internal force proportional to velocity of any dependent form parameter) can be taken into account by

$$\sigma = k \cdot \dot{\varepsilon} \tag{8.19}$$

in which k is the viscous damping coefficient. The equilibrium forces of the element according (8.10), with negative sign (reaction forces), can be used as external forces acting at the co-ordinates of the element.

8.4.3 Gravity forces

Actually these forces concern mass forces in the acceleration field of the gravity. A useful idea is to define a gravity vector $\underline{g}$ concerning all co-ordinates of the mechanism model. For the usual gravity in the –y direction such a vector would have coefficients –g in all y-co-ordinates. Such a vector will typically look like (g without underline is the gravity constant):

$$\underline{g} = |\ 0 \quad -g \quad 0 \quad -g \quad 0 \quad -g \quad \cdots\cdots|^T \tag{8.20}$$

The (external) gravity forces can be calculated then using the mass matrix M of the mechanism:

$$\underline{f}_z = [M] \cdot \underline{g} \tag{8.21}$$

Notice that the gravity forces act at all co-ordinates, even the fixed ones. The latter forces contribute to bearing forces, but only in an unambiguous way when the bearing has been modelled properly (see 8.2.4).

8.4.4 Other external forces (Process forces)

Mechanisms will be designed to transfer motion and forces. Usually some process is to be carried out in which forces are needed. Some examples can be found in chapter 1, like in the cutting of steel blade, the digging of iron ore, the gas compression in a combustion engine etc. Such forces will be considered here as remaining external forces. They need a user specification.

8.5 Equation of motion

8.5.1 General principle

The equilibrium equation (8.6) can be used again, but now with the mass forces and other external forces:

$$\sigma^p = \left[D^{pcT}\right]^{-1} \cdot \left\{ f^c - M^c(\ddot{x}^c - g^c) \right\} \tag{8.22}$$

The matrix M^c is the part of the total mass matrix dealing with the moving co-ordinates (shaded part in fig. 8.3.5).

Vectors $\ddot{x}^c$ and g^c concern the moving co-ordinates of the acceleration vector and the gravity vector (eq. 8.20) respectively.

In vector f^c all other external forces (also from springs and dampers) are contained here.

The matrix D^{pc} concerns the first order continuity equations, the kinematic relation between prescribed form parameters and moving co-ordinates of the mechanism.

The accelerations as expressed in (5.30) with transfer functions and driving functions (of time)

$$\ddot{x}^c = \dot{\varepsilon}^{mT} . x'^{c} . \dot{\varepsilon}^m + x'^{c} . \ddot{\varepsilon}^m \tag{8.23}$$

can be substituted in the equilibrium equation. Now this equation (8.22) is a (differential) equation in the driving quantities ε^m. This equation will be called further the ***equation of motion*** of the mechanism

For practical use the part of the equation of motion, which concerns the input of the mechanism, deserves special attention. This part can be written separated as

$$\sigma^m = x'^{cT} \cdot \left\{ f^c - M^c(\ddot{x}^c - g^c) \right\} \tag{8.24}$$

Together with (8.23) this part is called further the ***equation of driving motion***.

The equation of motion can be used in the following two cases.

- All input motions $\varepsilon^m(t)$ are given functions of time, and so are the input velocities and accelerations. In any mechanism position the accelerations of all co-ordinates can be calculated then according (8.23). The equation of motion provides the internal forces directly, including the driving forces. This type of calculation will be called here ***dynamic analysis***.
- The driving forces σ^m are given. Now the equation of driving motion is a differential equation in ε^m, which must be solved first. Once $\varepsilon^m(t)$ and its time derivatives are known, the first case can be followed. There is no objection to specify the driving force as a function of the input motion:

  $$\sigma^m = \sigma^m(\varepsilon^m, \dot{\varepsilon}^m, \ddot{\varepsilon}^m) \tag{8.25}$$

  Such a specification is called further a ***motor model.*** This type of calculation is known also as ***(dynamic) simulation***.

8.5.2 Reduction of forces to the driving shaft

Assume here a mechanism with one degree of freedom α. Substitution of (8.23) in (8.24) leads to

$$\sigma^m = -\left[\left\{ x^{/cT} M^c x^{/c} \right\} \ddot{\alpha} + \left\{ x^{/cT} M^c x^{//c} \right\} \dot{\alpha}^2 - x^{/cT} \cdot (f^c - M^c g^c) \right] \tag{8.26}$$

In the literature a comparable equation can be found:

$$T_\alpha = J_{red} \ddot{\alpha} + \frac{1}{2} \frac{dJ_{red}}{d\alpha} \dot{\alpha}^2 - f_{red} \tag{8.27}$$

J_{red} is called the inertia moment of the mechanism reduced to the input shaft. F_{red} contains the reduced external (other) forces, including gravity forces. The comparable terms in the FEM-approach can be recognized easily. The mass matrix is usually a constant matrix, but the first and second order transfer functions are typically not constant. The reduced inertia is thus dependent on the mechanism position. Notice that (8.26) shows very nice how kinematics plays a role in dynamics.

To find the *static moment (brake moment*) to hold the mechanism immovable eq.(8.27) defaults to

$$\sigma^m = x^{/cT} \cdot f^c \tag{8.26}$$

which has been found earlier in chapter 8.2.2, see eq.(8.7). This result is also valid for multi-DOF mechanisms (σ^m is a vector then and $x^{/c}$ a matrix). Writing out the equation (8.26) for a multi-DOF mechanism will be omitted here, both eq. (8.23) and (8.24) together express the equation correctly.

8.5.3 Driving force and motor model

Examples of motor models:

- A constant force (for instance a braking force to stop a running mechanism).
- A zero force (for instance the mechanism will be set free at a certain time and starts moving due to the external forces).
- A spring motor (linear relation between σ^m and ε^m), compare eq. (8.17).
- An asynchronous AC-motor with a steady-state characteristic as depicted in fig. 1.4.1 (left part). Here the driving moment $M = -\sigma^m$ is a function of the angular speed $\dot{\alpha} = \dot{\varepsilon}^m$. In the literature a mathematical expression for such a motor model can be found:

$$M = 2\kappa M_{max} \frac{s}{1 + \kappa^2 s^2} \quad ; s = \frac{\dot{\alpha} - \dot{\alpha}_{max}}{\dot{\alpha}_{max}} \quad ; \kappa = 2 \tag{8.29}$$

In design practice the *choice of a driving motor* may need both applications of the motion equation. The selection of the driving motor (out of a catalogue) will typically be based on worst-case assumptions, like maximum required power, maximum required torque at a certain (maximum) speed etc. In a machine concept with a central driving shaft it will be acceptable then to suppose a constant speed of the driving shaft. In a multi-DOF robotic mechanism the given motion functions and process forces will be applied at the “wrist” of the robot (motion task) and need to be transformed first to the input motion to find the maximum required driving torques (using the inverted kinematic model). Once a driving motor has been selected, the properties of the motor are known. The simulation can be used for instance to verify whether or not the driving speed is constant enough or the robot task will be performed accurate enough.

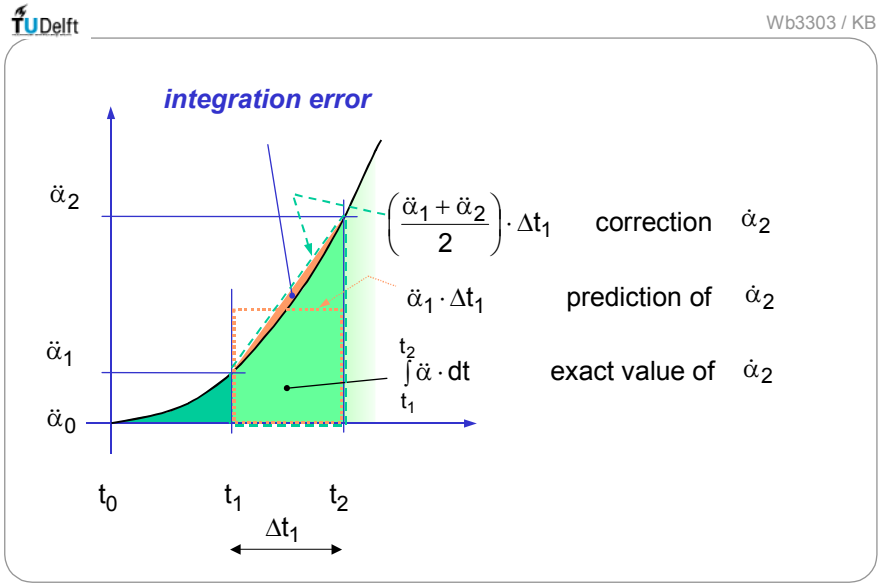

Fig. 8.5.1 *Integration error in the equation of motion for the crank angle α*

8.5.4 Integration of the driving equation of motion

For dynamic simulation the equations of driving motion need to be solved (integrated). Since the coefficients depend on the mechanism position (the variables of the equation to solve) they are non-constant. Only numerical integration methods come into account.
Standard procedures for numerical integration are well known (like method of Heun, method of Runge/Kutta, method of Gordon and Shampine and others). To give an impression of the required calculations and some problems of the integration procedures, an example will be presented here (Heun).

Assume a one-DOF mechanism with no driving motor attached at the input shaft (the mechanism will move freely when released at start time). The driving force is thus zero.

At start (time t_0) the input position is α_0 and the velocity $\dot{\alpha}_0 = 0$. In this position the acceleration of the input shaft can be calculated with (8.26), using the kinematic first order transfer functions, the external forces and the mass matrix:

$$\ddot{\alpha}_0 = \frac{f_{red}}{J_{red}} = \frac{x_0^{/cT} f^c}{x_0^{/cT} M\, x_0^{/c}}$$

It will be clear that the mechanism will start to move only when at least one of the external forces is non-zero.

After some small time step Δt, at time $t_1 = t_0 + \Delta t$, the new position and velocity can be ***predicted***:

$$\begin{aligned} \alpha_1 &= \alpha_0 + \dot{\alpha}_0 \Delta t + \tfrac{1}{2} \ddot{\alpha}_0 \Delta t^2 \\ \dot{\alpha}_1 &= \dot{\alpha}_0 + \ddot{\alpha}_0 \Delta t \end{aligned} \qquad (8.30)$$

and (8.26) can be used to calculate the acceleration of the input shaft at t_1:

$$\ddot{\alpha}_1 = \frac{x_1^{/cT} f^c - x_1^{/cT} M\, x_1^{//c}\, \dot{\alpha}^2}{x_1^{/cT} M\, x_1^{/c}} \qquad (8.31)$$

Now it should be verified how accurate the integration is. This can be done by recalculating $\dot{\alpha}_1$ and α_1, reading for i the position 1:

$$\begin{aligned} \dot{\alpha}_i &= \dot{\alpha}_{i-1} + \tfrac{1}{2}(\ddot{\alpha}_{i-1} + \ddot{\alpha}_i)\Delta t_i \\ \alpha_i &= \alpha_{i-1} + \tfrac{1}{2}(\dot{\alpha}_{i-1} + \dot{\alpha}_i)\Delta t_i \end{aligned} \qquad (8.32)$$

The difference between (8.30) and (8.32) is the integration error on $\dot{\alpha}_1$ and α_1 respectively, see fig. 8.5.1 for the error on $\dot{\alpha}$.
The ***correction cycle*** can be repeated until this error is small enough. The precise amount is dependent on the accuracy demands of a user. When the error remains too big, a smaller time step may improve the solution.

Further remarks:

- The integration requires that the denominator in eq. (8.31) is non-zero. The mechanism must have some moving mass!
- It is well known that such numerical integration procedures may become divergent (unstable). One reason could be that an external force is not continuous. In that case the predicted acceleration (8.29) will also have the jump. Bad convergence can be the result. Typically the use of a triple spring as proposed in fig. 8.4.2 is risky in simulation.

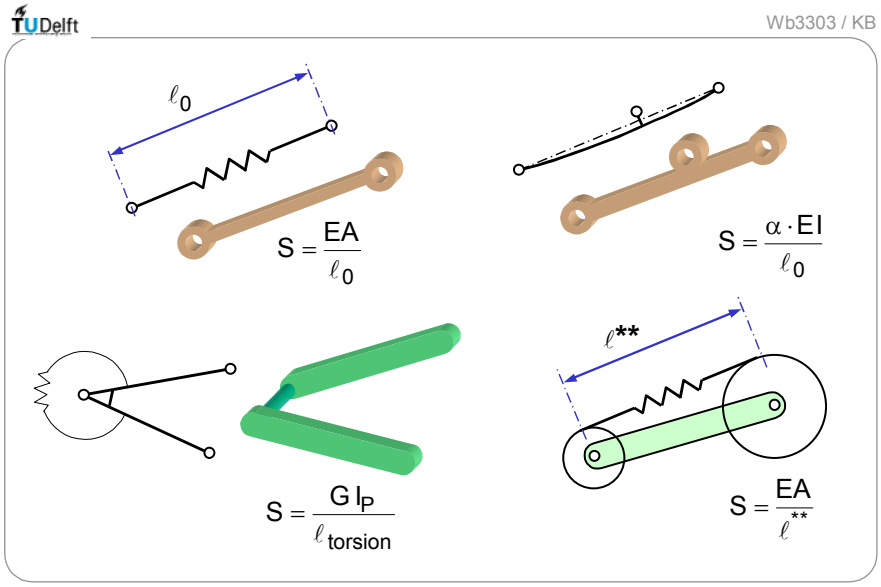

Fig. 8.6.1 Some stiffness interpretations of elements

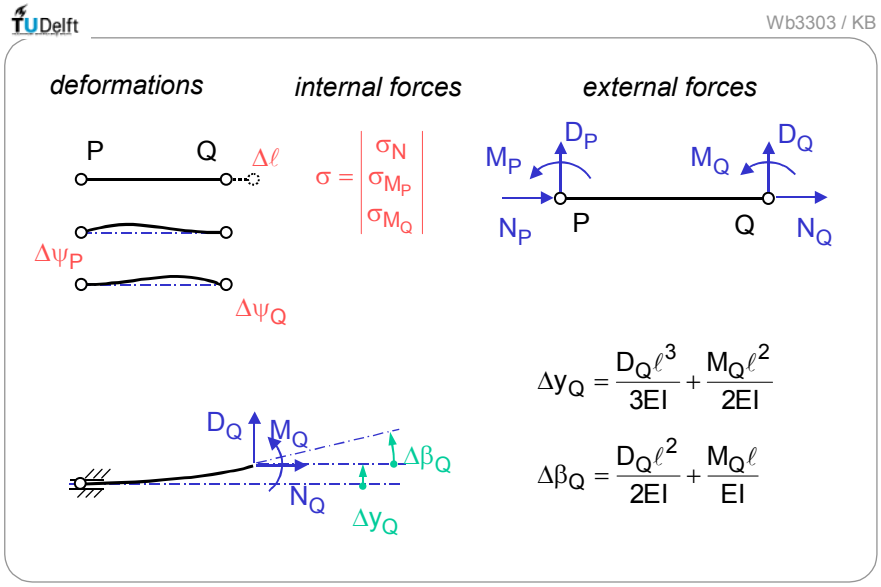

Fig. 8.6.2 Coupled bending stiffness of the beam element

8.6 Stiffness of the elements and deformations

8.6.1 The constitutive equations

It will be assumed that the deformations are small and proportional to the forces. The deformations (due to forces) in the FEM approach concern the ***prescribed form parameters***. In kinematics they are rigid (prescribed zero), in dynamics they can be given a finite stiffness by means of a so-called constitutive equation. Stiffness (S) is a property of an element, so the equation can be described for each element k separately:

$$\sigma^k = S^k \cdot \Delta\varepsilon^k \tag{8.33}$$

The stiffness must be understood as ***generalized***. It depends on the type of the form parameter how the stiffness is defined. Some typical examples are depicted in fig. 8.6.1. Sometimes the user can give his own interpretation to the stiffness, with respect to the constructive realisation of the links.

In most elements presented so far, the deformations within one element are ***decoupled***, that means there is one equation for each prescribed form parameter. This is even true when two elements are assembled, like in the fixed angle (by means of a torsion bar) between two links, see fig. 8.6.1 left below. The deformation can then, for given internal force, be expressed as:

$$\Delta\varepsilon^k = \frac{\sigma^k}{S^k} \tag{8.34}$$

There is however one exception: the beam element, see the next chapter.

8.6.2 Bending of the beam element

The two bending deformations at the end-points should be coupled to define proper bending deformation, see fig. 8.6.2. To find the deformations of the whole element (assume also the length is prescribed and has normal stiffness) the element can be placed in an arbitrary but suitable position (horizontal in fig. 8.6.2). In this position the deformations are related to the displacements of the co-ordinated by the continuity equations (as used in kinematics):

$$\Delta\varepsilon^k = D^k \Delta x^k$$

$$\begin{vmatrix} \Delta\ell \\ \Delta\psi_P \\ \Delta\psi_Q \end{vmatrix} = \begin{vmatrix} 1 & 0 & 0 \\ 0 & -\frac{1}{\ell} & 0 \\ 0 & +\frac{1}{\ell} & -1 \end{vmatrix} \cdot \begin{vmatrix} \Delta x_Q \\ \Delta y_Q \\ \Delta\beta_Q \end{vmatrix} \tag{8.35}$$

Standard formulas are available to express the displacements of the co-ordinates (only for point Q, since point P is kept fixed):

$$\Delta x^k = S^{*k} f^k$$

$$\begin{vmatrix} \Delta x_Q \\ \Delta y_Q \\ \Delta\beta_Q \end{vmatrix} = \begin{vmatrix} \frac{\ell}{EA} & 0 & 0 \\ 0 & \frac{\ell^3}{3EI} & \frac{\ell^2}{2EI} \\ 0 & \frac{\ell^2}{2EI} & \frac{\ell}{EI} \end{vmatrix} \cdot \begin{vmatrix} N_Q \\ D_Q \\ M_Q \end{vmatrix} \tag{8.36}$$

Assume now the internal forces σ (σ_N, σ_{MP} and σ_{MQ}) are given. Eq.(8.36) and the equilibrium equations (8.10) substituted in (8.35) yields:

$$\Delta\varepsilon^k = \left\{ D^k \cdot S^{*k} \cdot D^{kT} \right\} \cdot \sigma^k = \left[S^k \right]^{-1} \cdot \sigma^k \tag{8.37}$$

The term in brackets is by definition the (inverse) stiffness matrix. Although the D^k matrix is dependent on the position of the element, the result is of course a constant matrix. Notice that it is not necessary to do the matrix inversion, it can be obtained directly by matrix multiplication as indicated. The final result is:

$$\left[S^k\right]^{-1} = \begin{vmatrix} \frac{\ell}{EA} & 0 & 0 \\ 0 & \frac{\ell}{3EI} & \frac{\ell}{6EI} \\ 0 & \frac{\ell}{6EI} & \frac{\ell}{3EI} \end{vmatrix} \tag{8.38}$$

in which the bending stiffnesses are coupled. It is a good idea to call this inverted stiffness matrix the flexibility matrix of the beam element. All stiffnesses are actually regarded as flexibilities.
Practically it means that rigidity (infinite stiffness) should be specified as zero flexibility. This is a better idea regarding numerical values in computer programs.

8.6.3 Position accuracy of the mechanism

For a given internal force distribution and given flexibility of the links the deformations of the prescribed form parameters can be calculated using (8.34) or (8.36). The displacements of the co-ordinates due to these deformations can be considered with respect to position accuracy.
They can be calculated with the continuity equations (matrix D^c) as used in kinematics for position update (5.18).
Actually the inverse of matrix D^c can be understood as the sensivity matrix for deformations. When multiplied with the deformation vector, the (linearized) co-ordinate displacements are the result. For precise calculation of the accuracy the iterative procedure could be used as described in chapter 5.3.4.

Remark: the deformations due to the mass forces can be less accurate (see chapter 8.3.5)

Typically such accuracy calculation will be used to find out the maximum deviations of rigid body motion. The linearized estimation of co-ordinate deviation provides usually sufficient information whether or not the element is stiff enough. It is a good idea to take here all prescribed form parameters as deformable, even if the flexibility is very low.

8.6.4 Dynamic simulation with flexible elements

To find out how the deformations of a mechanism behave as a function of time, a dynamic simulation is needed. There is no new theory required. The mechanism model can be adapted to do this task.
The basic idea is that any elasticity must be regarded (modelled) as a degree of freedom (drive) at which a driving characteristic can be added. Here the spring characteristic comes into account as the motor model. To avoid confusion such a flexibility will be called a dynamic degree of freedom, and the original ones the kinematic degrees of freedom.

Remark: do not confuse with a spring (a spring does not change the degree of freedom)

Dynamic degrees of freedom need not to describe large deformations. Bending deformation of the beam element is typically suited as a dynamic degree of freedom.
With respect to the integration of the equations of motion, in addition to chapter 8.5.4 it can be remarked:

- The use of flexibility, without damping, will lead to undamped oscillating motion. When the external forces vary with about the same frequency, the oscillations may become very large.
- A high frequency of the oscillation (high stiffness or low mass) needs a small stepping time in the integration procedure, which will become slow then.

```
DYN ANA   ;test example dynamics (for Runmec version 3.x)
TOPOLOGY
 elem 1  BIN  1  2 18 19      ; support of point A0
      2  BIN  3  4 18 19      ; support of point A0
      3  BIN 18 19 10 11
      4  TR2 10 11 12 13 16 17 ; extra mass point on the coupler
      5  BIN  5  6 14 15
      6  TR1 12 13  9 14 15   ; bearing in point B
      7  BIN  7  8 14 15      ; spring
 form 1 lenl  1   1  1
      2 lenl  2   2  1
      3 lenl  3   3  1
      4 lenl  4   4  1
      5 lenu  4   5  1
      6 lenv  4   6  1
      7 lenl  5   7  1
      8 lenu  6   8  1
      9 lenv  6   9  1
     10 angb  3  10  1
     11 lenl  7  -1  1  ; dependent form parameter (spring)
nrdof 1
GEOMETRY
 xfix  1   0.0
       2  -0.1
       3  -0.1
       4   0.0
       5   0.4
       6   0.0
       7   0.7
       8   0.0
       9   0.0

 xmov 10  -0.1
      11   0.2
      12   0.4
      13   0.5
      14   0.4
      15   0.5
      16   0.3
      17   0.4
      18   0.0
      19   0.0

para   4   0.6
       5   0.3
       6   0.0
DYNAMICS
 inertia  16  0.5
          17  0.5
 uniform   3  1.3
           4  2.6
           5  1.7
 spring   -1  200 -50  ; c and sigma at length zero
 gravity 9.81  -1.5708
STORAGE
 bufinp     1  100  ; crank angle
 bufx0     16 101  ; coupler point
 bufx0     17 102
 bufdx2    16 105  ; acceleration of coupler point
 bufdx2    17 106
 bufe0     -1 103  ; length of the spring
 bufsigma   1  201  ; normal force in link 1 (vertical support)
            2  202  ; normal force in link 2 (horizontal support
            6  206  ; mass force at coupler
            8  208  ; horizontal bearing force in B
            9  209  ; vertical bearing force
           10  210  ; driving force
 PRINTBUF
 ANIMATE film
 BEGINPIC
  plotcurv 101 102  ; coupler curve
  drawmech 1
 ENDPIC
 BEGINPIC
  plotcurv 202  201 ; polar curve of support force in A0
  plotcurv 208  209 ; idem of bearing forces in point B
 ENDPIC             18 15 0 0 -1 -2 ;(autom. scaling, true distance)
MOVEMENT
 tstepe 120 0.20944 30.0 0.0  ; exactly one cycle
END
```

Table 8.1

It will be clear that the amount of calculations rapidly increases with the number of dynamic DOF's. It is recommended then to carefully make use of flexibilities in simulation calculation. Typically only few flexibilities should be considered, and only in case the flexibility is relatively high.

8.7 Example with RUNMEC

The mechanism of fig. 8.7.1 has been analysed dynamically (kineto-statics). The mechanism has been given several dynamic properties, like:

- A spring (element 7)
- A coupler with bending force (element 4)
- Support element of the fixed point A_0
- Bearing forces at point B

The input file for the RUNMEC program is listed in table 8.1. The output plot of the coupler curve is presented in fig. 8.7.2. The polar plots of the bearing forces (in A_0 and B) have been depicted in fig. 8.7.3.

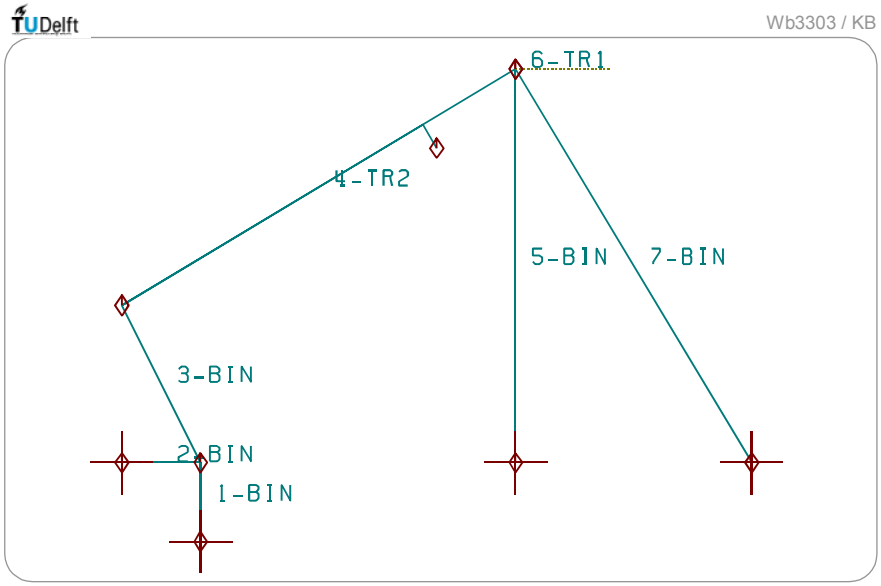

Fig. 8.7.1 Sample mechanism for dynamic analysis

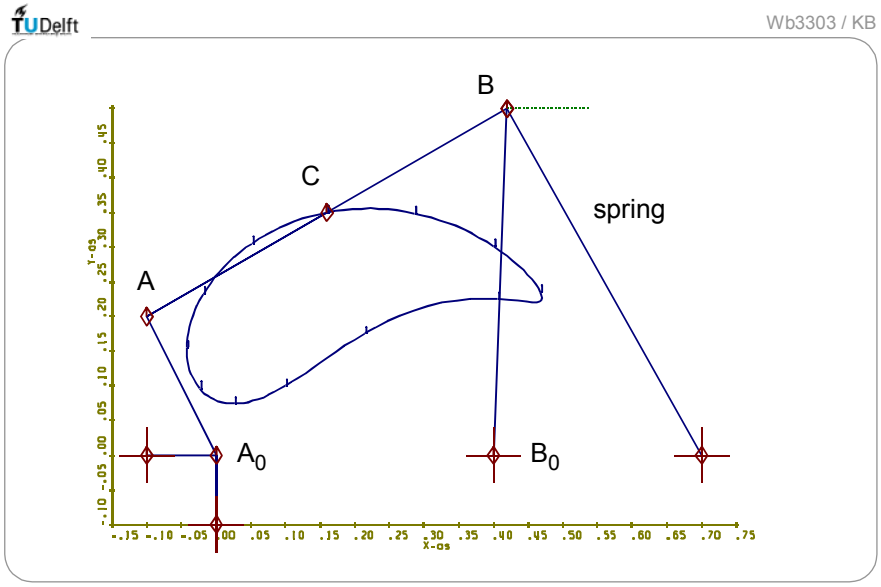

Fig. 8.7.2 Mechanism and coupler curve

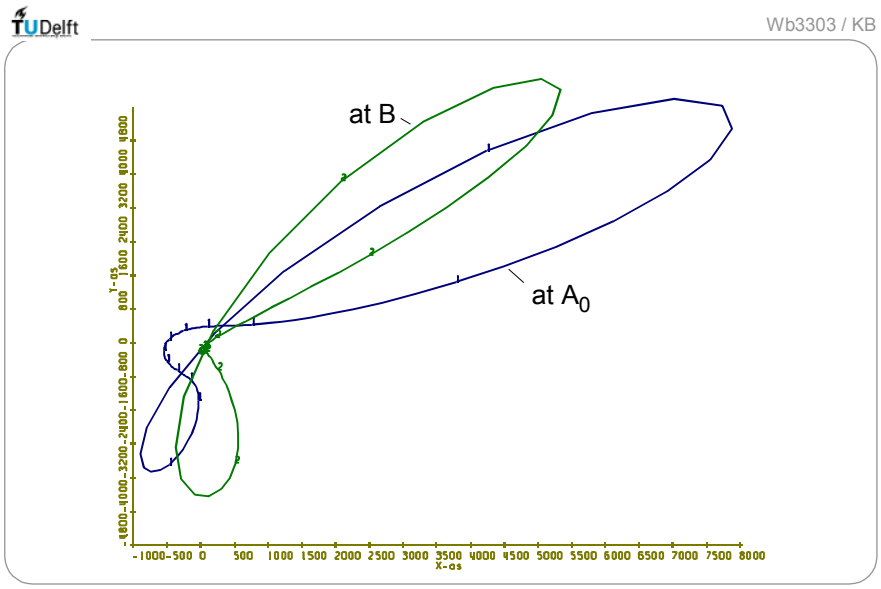

Fig. 8.7.3 Polar plots of bearing forces