MEMORANDUM M-843

Path-following Methods for Contact Analysis

G.J. van Tilburg

March, 1998

Delft University of Technology
Faculty of Aerospace Engineering
Delft, The Netherlands
Path-following Methods for Contact Analysis

G. J. van Tilburg
Faculty of Aerospace Engineering
Department of Computational Mechanics (C2)
Delft University of Technology
exam committee:

Prof. dr. J. Arbocz  
Delft University of Technology, Faculty of Aerospace Engineering,  
Department of Computational Mechanics

ir. A.W.A. Konter  
MARC Analysis Research Corporation Europe, Zoetermeer

Dr. ir. E. Riks  
Delft University of Technology, Faculty of Aerospace Engineering,  
Department of Computational Mechanics

This report contains 72 pages
summary

The use of pathfollowing methods in contact problems requires special adaptation in order to work properly. This report describes the implementation of such a method in the MARC code. The special points of consideration are the discontinuous points that are encountered in the continuation when contact occurs and the problem of the changing set of degrees of freedom. Also discussed are two particular implementations of the arclength method that have a marked influence on the robustness of the resulting algorithm.
samenvatting

Dit verslag behandelt de pad-volgingsmethode voor kontaktproblemen in niet lineare quasi-statische constructieve problemen, met gebruik van het eindige elementen programma MARC. Met de pad-volgingsmethode is het mogelijk de oplossingskromme stapsgewijs te volgen. De methode is een incrementele iteratieve methode. In het programma van MARC zijn twee pad-volgingsmethoden geïmplementeerd, de Riks methode en the Crisfield methode. Na een afleiding van de standaard methode zullen deze twee methoden vergeleken worden.

Om de problemen in het gebruik van de pad-volgingsmethoden voor kontaktproblemen te kunnen moet de modellering van kontakt in MARC bestudeerd worden. Deze methode verschilt van de methoden zoals die veelal in de literatuur behandelt worden voor het oplossen van kontakt problemen, en is gebaseerd op het "vastknopen" van knooppunten wordt genoemd. De "tyings" zijn betrekkingen waarin de verplaatsingen van de knopen van het ene contactlichaam worden gekoppeld aan de verplaatsingen van de knopen van het andere contactlichaam. De "tyings" zorgen voor een reductie van het aantal vrijheidsgraden als kontakt optreedt. Deze reductie zal blijken geen invloed te hebben op de relaties voor de pad-volgingsmethode wanneer het systeem van evenwichtsvergelijkingen terug getransformeerd wordt naar het systeem met alle vrijheidsgraden.

Kontakt zorgt voor discontinuïteit in the afgeleide van de oplossingskromme, en dus, voor de richting waarin de oplossing zich begeeft. Vanwege de discontinuïteit moet de analyse gesplitst worden op het moment dat een knoop van het ene contact lichaam het andere raakt waarna de stijfheidsmatrix opnieuw berekend kan worden. De stapsgewijze benadering met ongeveer constante stappen wordt behouden als een increment wordt gesplitst in subincrementen. Deze onderbrekingen van de analyse kan ook gebruikt worden om en nieuwe stappgroote te bepalen na kontakt. Discontinuïteit van de afgeleide kan problemen geven bij de definitie van het systeem na kontakt, tenzij het controle oppervlak van de iteratie veranderd wordt. In dat geval wordt singulariteit vermeden.
preface

This is a report of the graduation project that concludes my study of Aerospace Engineering at Delft University of Technology. The study is performed of behalf of MARC Analysis Research Corporation, Zoetermeer.

When I started my graduation project at MARC A.R.C. Europe the last version of the program was version MARC K6.2. With this version one was able to compute nonlinear contact problems as long as the externally applied loads or displacements were prescribed by fixed steps. For nonlinear problems without contact, an automatic stepping procedure was available but this procedure did not work properly with contact. The analysis of this problem was the basis of the project and the study in this report. This improved option is now available in the latest version MARC K7.2.

This report intends to be a reference for those who are interested in a derivation of the basic path-following method for nonlinear problems. It is extended with a procedure to solve discontinuous points in the path due to contact. Although the discussion is about discontinuities due to contact, it can be used for all kinds of discontinuities in the path. Furthermore this report can be used by the MARC users as a reference for the use of the auto increment procedure for nonlinear problems with or without contact.

I would like to thank MARC for the possibility to perform this study. The people at the MARC office in Zoetermeer for all their assistance for gaining experience with MARC and the Mentat program, and the pleasant time I had there. Of major importance for the practical part of the study was Albert and I would thank him for his supervision and the nice and useful discussions we had. Cees and Adrie for their assistance at the programming part of the project. I would also like to thank the members of the exam committee Prof.dr J. Arbocz, ir. A.W.A. Konter and dr.ir E. Riks. dr.ir. E. Riks especially for his patience, guidance, advice and supervision on the theoretical part of the study. Obviously, I would like to thank my housemates, my parents and my fiancée Annelize for the mental support. My father and Arjan also for their assistance in this report.
Contents

1 Introduction ............................................................................................................. 13
  1.1 Objectives of this study .................................................................................... 13

2 Contact in FEM ...................................................................................................... 15
  2.1 Variational formulation .................................................................................... 15
  2.2 Formulation of the contact conditions ............................................................... 16
    2.2.1 Normal Contact conditions ........................................................................ 18
    2.2.2 Sliding and friction .................................................................................... 19
    2.2.3 Discretization of the contact conditions ..................................................... 20
  2.3 Imposition of constraints ................................................................................ 24
    2.3.1 Lagrange multiplier method ....................................................................... 24
    2.3.2 Penalty method ......................................................................................... 24
    2.3.3 Tying method ............................................................................................. 25
    2.3.4 Solving the tyings ...................................................................................... 27
  2.4 Contact search .................................................................................................. 30
  2.5 Aspects of contact analysis ............................................................................... 32

3 Path-following methods ....................................................................................... 33
  3.1 Incremental approach ...................................................................................... 33
    3.1.1 Path-following process .............................................................................. 38
  3.2 Solving the non-symmetric system ................................................................... 39
  3.3 Path-parameter ................................................................................................ 42
  3.4 Implementation of the path-following method in MARC ................................ 45
    3.4.1 Riks’ implementation of path-following .................................................. 45
    3.4.2 Crisfield’s implementation of path-following ............................................ 48
  3.5 Conclusions ..................................................................................................... 50

4 Continuation methods for contact analysis ........................................................ 51
  4.1 The tying relations in path-following methods ............................................... 51
    4.1.1 Nonhomogeneous tying relations ............................................................... 53
  4.2 Discontinuity of the path .................................................................................. 55
    4.2.1 increment splitting ..................................................................................... 56
    4.2.2 Increment splitting in the path-following methods .................................... 58
  4.3 Redefinition of the auxiliary surface ................................................................. 59
  4.4 Considerations about the steplength ................................................................ 61
  4.5 Problems of large residuals in the implementations ........................................ 61
5 Conclusions .................................................. 63
  5.1 Conclusions .............................................. 63
  5.2 Recommendations ..................................... 63

A1 Geometrical interpretation of the path-following method ............... 67
  A1.1 Geometrical interpretation of the linearizations ................. 67
  A1.2 Further considerations .................................. 68

A2 Path-following procedures .................................... 69
  A2.1 Procedure of Riks’ implementation .......................... 69
  A2.2 Procedure of Crisfield’s implementation .................... 70

B Demonstration of problems in contact .................................. 71
List of symbols

\( \delta A \)  
virtual work

\( A \)  
matrix for the projection of the hitting nodes on the target nodes

\( E \)  
tensor of elasticity

\( f(v) \)  
system of equilibrium equations

\( f_{N+1} \)  
augmented equation

\( F(v) \)  
augmented system of equilibrium equations

\( g,g \)  
gap and vector of gaps

\( h(v) \)  
function of the auxiliary surface

\( J \)  
Jacobian matrix

\( K \)  
stiffness matrix

\( l \)  
load vector

\( L \)  
displacements due to the load

\( m \)  
normal to the auxiliary surface

\( n \)  
normal

\( N \)  
matrix of interpolation functions

\( p \)  
surface traction

\( S \)  
second Piola-Kirchhoff stress tensor

\( r \)  
residual force vector

\( r_0 \)  
residual to the auxiliary surface

\( R \)  
displacements due to the residuals

\( t \)  
tangent to the contact surface

\( u \)  
vector of nodal displacements

\( \hat{u} \)  
reduced displacement vector

\( \hat{u} \)  
displacement component in local direction

\( v \)  
vector of degrees of freedom

\( w \)  
displacements vector

\( x \)  
position vector

\( z \)  
unit local contact surface vector

\( \eta \)  
path-parameter

\( \eta \)  
Green-Lagrange strain tensor

\( \lambda \)  
load parameter

\( \Lambda \)  
vector of Lagrange multipliers

\( \mu \)  
fraction coefficient

\( \phi \)  
isoparametric interpolation function

\( \Phi \)  
prediction function

\( \Omega \)  
itration function
Chapter 1

Introduction

Finite element procedures (FEM) are at present very widely used in engineering analysis. This use can be expected to increase significantly in the years to come. Because of the increasing possibilities and the decreasing prices of the computers, the costs for the use of FEM is substantially reduced. Consequently, the growth of the number of users has been steady. This increase of the users is also caused by the demanding and increasing critical attitude of their customers and the trend to reduce the costs of production. The reduction of the cost can be achieved e.g. when costly pre-productions or test series can be avoided by simulation of the processes. The critical attitude of the customers expresses itself in severe requirements for the products and these requirements can be obtained, among others, by means of computations with e.g. FEM. The increasing demand for FEM packages has resulted in an increased supply of these packages as well. This and the critical attitude of the users make that the competition between the different FEM packages increases. The competition forces the developers to create more functionality, and facilitate the application of these methods to problems with more complexity.

Because of the increasing possibilities of the computers, the problems that can be solved numerically are of increasing complexity. The problems that one was enable to solve, in the early days of FEM practice, were linear and simplified. At present however, the demand for solvers for all kind of nonlinear problems is extended. After all, nature is nonlinear. An important class of nonlinear problems is the class of contact problems. Despite of its importance e.g. for the simulation of metal forming and impact, the study of contact problems is rather young, mainly due to the highly nonlinear nature of these problems. Contact problems can be divided in static and dynamic contact. Dynamic contact is often called impact and is of importance for the simulation of e.g. crash tests. The discussion in this report is restricted to static equilibrium of contacting bodies.

The path-following method as a general tool for the solution of nonlinear problems in solid mechanics was at first considered by Riks (1970). The use of path-following techniques has steadily increased and many nonlinear finite element methods are now equipped with it. Because of this attention, many modifications to the basic derivation of Riks have been proposed. Most of these modifications, however, are just slight variations and, therefore, in this report the derivation according to Riks is followed. The use of the path-following method for contact analysis is quite new and is as expected accompanied with some new difficulties that will be considered in this report.

1.1 Objectives of this study

This study is performed in behalf of MARC Analysis Research Corporation and consequently, it will be based on the problems and the formulations of the implementation in the MARC-code. Furthermore, the notation used here is a mixture of the notation used in MARC and the notation which can be found in the literature. The assignment for this study has been formulated as follows: Review and test path-following methods available for automatic load stepping in the nonlinear behaviour of structures. Particular emphasis should be paid to the combination of contact phenomena and the automatic load stepping procedure. As a consequence, the study is intended to give a detailed description of the path-following methods. Other options to solve nonlinear problems are not considered.
Before the study has been started only one continuation method was implemented. This method, attributed to Crisfield, has some drawbacks and therefore a second method is implemented. The latter procedure, which was based on the theory introduced by Riks, is reviewed and compared with Crisfield's method.

The second part of the study is intended to solve the difficulties that occur when the path-following method is used for contact analysis. This discussion will be restricted to static equilibrium of contacting bodies. To do this properly, it is necessary to know the way contact is modelled in the MARC-code. The approach in MARC is different from the commonly used methods and is based on the so-called tyings of nodes. To illustrate the differences, the other methods are also briefly considered. The main difficulties of using the path-following method for contact analysis are firstly, the implementation of the tyings in the relations for continuation of the path, and secondly, the procedure to deal with discontinuous points in the path when bodies are contacting. Because the path-following method according to Riks turns out to be the most simple and stable, the discussion about the problems in contact analysis is focussed on this implementation.
Chapter 2

Contact in FEM

2.1 Variational formulation

A general way to derive the equations that govern the behaviour of the structure is to use the principle of virtual work. The exposition in this chapter is based on this approach. For a deformable structure, under a certain external set of loads, the virtual work principle can be stated as: A general deformable body is in equilibrium if, and only if, the total virtual work done by the internal stress field and the external forces is zero, for every virtual displacement field consistent with the geometrical constraints.

The virtual work can be written as an expression in terms of the stresses, displacements and the externally applied load. The second Piola-Kirchhoff stress tensor $S$ is physical dual to the Green-Lagrange strain tensor $\eta$, and therefore, when their scalar product is integrated over the body $V$, the work done by the internal stressfield is obtained. It is assumed that the configuration is only loaded on its boundary $S$ (body forces are neglected). The work of the externally applied load can then be expressed as the scalar product of the externally applied surface traction $p$ and the displacement vector $w$ integrated over $S$. Because proportional loading is assumed, the load is written as $p = \lambda p^0$, where $p^0$ is the nominal load. Herewith, the virtual work can be expressed by

$$\delta A = \int_V S \cdot \delta \eta dV - \lambda \int_S p^0 \delta w dS = 0 \quad 2.1$$

where $\delta \eta$ are the variations of the strain and $\delta w$ are the variations of the displacements. Notice that $S$ and $\eta$ are second order tensors. In general linearly elasical material, the second Piola Kirchhoff tensor is dependent of the strain tensor $\eta$ and the Elasticity tensor of the material. This tensor of elasticity, $E$, is a fourth order tensor. When the strain tensor $\eta$ is written in terms of the derivatives of the displacements, $\eta = \eta(w)$, the second Piola Kirchhoff stress tensor can be written as

$$S = E \eta(w) \quad 2.2$$

When the continuous deformable body is discretized, the body is divided into elements. The displacements within an element are determined by interpolation of the displacements of the adjacent nodes. Accordingly, the displacement vector in a material point, in the three global directions, $w$, can be written in terms of the nodal displacement vector $u$ in the global directions, using the isoparametric interpolation functions $N$. The displacements in the global directions for a point with coordinate vector $x$ are transformed by

$$w = N(x) u \quad \quad \quad \quad \quad N \in R_3 \times R_{3N} \quad 2.3$$
where $N$ is the number of nodes. Consequently, the strain tensor can be written in terms of the nodal displacements, $\eta = \eta(u)$. Substitution of (2.3) and (2.2) in the virtual work expression (2.1) gives

$$\delta A = \int_V \mathbf{E} \delta u \cdot \delta \eta dV - \lambda \left( \int_S \mathbf{P}^0 \delta (N(x) u) dS \right) = 0 \quad \text{(2.4)}$$

With the orthogonal unit base vectors denoted by $e_i$, the components of the displacements by $u_i$, the displacement $u$ is given by $u = u_i e_i$, where summation over the index (i) is implied. In component notation, the virtual work expression (2.4) can then be given by:

$$\delta A = \int_V \eta_{ij} E_{ijkl} \frac{\partial \eta_{kl}}{\partial \omega_{ik}} N_{mn} \delta u_n dV - \lambda \left( \int_S \mathbf{P}_k^0 N_{kn} \delta u_n dS \right) = 0$$

$$= \left\{ \int_V \eta_{ij} \tilde{E}_{ijkl} N_{mn} dV - \lambda \left( \int_S \mathbf{P}_k^0 N_{kn} dS \right) \right\} \delta u_n = 0 \quad \text{(2.5)}$$

$$= \delta u^T f(u, \lambda) = 0$$

This yields:

$$f(u, \lambda) = 0 \quad \text{(2.6)}$$

as the notation for the equilibrium equations.

### 2.2 Formulation of the contact conditions

Contact between two bodies can be defined by constraints. These constraints can be defined in the normal and the tangential direction relative to the surface of a contacting body. The normal condition is the physical requirement that the bodies are unable to penetrate each other. Contact in tangential direction can be divided into two phenomena, sliding and friction. Sliding is the movement tangential to the contact surface, and friction is the natural resistance against it.

The contact conditions, for contact of continuous deformable bodies, are transformed in constraints for the nodal displacements, so that they can be implemented in FEM. To show this, first the virtual work for contact is considered and next its discretisation. However, the discussion about the discretization of the contact conditions is kept brief because details can be found in the literature e.g. in Zongh (ref. (17)). The discussion is restricted to two deformable bodies in 3D but it is not difficult to generalize the concept to multiple body contact or even to self contact (Zongh). The mathematical foundation of the contact analysis can be found e.g. in Kikuchi and Oden (ref. (5)).

To discuss the virtual work expression for problems with contact, firstly, contact must be described. Consider two deformable bodies, I and II, with corresponding surfaces, $S^I$ and $S^II$. Body I and II are denoted as the hitting and the target body, respectively. For both surfaces a local orthogonal coordinate system ($z_1, z_2, z_3$) is introduced and the directions are defined as follows: The direction denoted by $z_3$ corresponds to the normal to the surface $n$, and is taken positive outward (for future reference, this direction is denoted as the third direction). The two other orthogonal vectors $z_1$ and $z_2$ are tangential. The vector product between the two mutually orthogonal tangential unit vectors defines the normal as
\[ n = \pm z_1 \times z_2 \]

The sign in this relation determines the outward direction but the details of this convention are considered later. When the surfaces of the bodies are assumed to be sufficiently smooth, the normals to the bodies in a point of contact are equal but are of opposite sign. This condition is given by

\[ n^I = -n^II \]

where the superscript indicates the surface \( S^I \) at the point of consideration. In the same way, it is possible to define for the tangential directions: \( z^I_i = -z^II_i \), for \( i = 1, 2, 3 \). The resulting local coordinate systems for the two bodies are thus oriented in an opposite way. Consequently, when the local coordinate system of the target surface is right handed, the coordinate system of the hitting surface must be left handed. This difference in orientation determines the sign in equation (2.7). This sign in the determination of the normal in equation (2.7) is determined by the definition that the normals are pointing outward and the following relation holds

\[ z^I_i = -z^II_i \quad i = 1, 2, 3 \]

Assuming that no sliding occurs, equality holds for the tangential surface tractions as well so that

\[ p^I = -p^II \]

This tractions can be decomposed in the local directions by \( \mathbf{b}_I = \mathbf{p} \cdot z_i \), and similarly, the displacements in the local directions by \( \mathbf{u}_I = \mathbf{u} \cdot z_i \) (\( \mathbf{p}_I \) and \( \mathbf{u}_I \) are preserved, respectively, for the components of the surface traction and displacement in the global directions). As long as two points from body I and II are in contact the displacements are equal. Consequently, the displacements \( \mathbf{u}^I_i \) and \( \mathbf{u}^II_i \) are equal but in the opposite direction.

\[ \mathbf{u}^I_i = -\mathbf{u}^II_i \]

---

**Figure 2-1 Demonstration of the contacting bodies for 2D**
Figure 2-2 Definitions for contact conditions

With this description of contact, the expression for the virtual work of the tractions by the virtual displacements of the contact surface, can be discussed. The second term of equation (2.1) is the external virtual work of the body forces and this term, rewritten in component notation, is given by

\[ \delta A_c = \int_{S} \hat{p}_i \delta u_i \, dS \]

\( \delta u_i \) is the variation of \( u_i \) and thus the virtual displacement. The actually contacting parts of the surfaces \( S^I \) and \( S^{II} \), are equal and denoted by \( \Gamma^I \) and \( \Gamma^{II} \). The expression for the virtual work of the contact traction, for virtual displacements of the two contacting surfaces, is given by

\[ \int_{\Gamma^I} \hat{p}_i \delta u_i^I \, d\Gamma + \int_{\Gamma^{II}} \hat{p}_i \delta u_i^{II} \, d\Gamma = \int_{\Gamma^{II}} \hat{p}_i \delta (u^{II} - u^I) \cdot z_i^{II} \, d\Gamma \]

For contact of a deformable body with a rigid body this relation can be simplified. The rigid body has no displacements except of the rigid body motion and thus does not have a contribution to the virtual internal work. When the deformable body is denoted with \( I \) and the rigid body by \( II \), the right hand side of equation (2.9) is reduced to

\[ \delta A_c = -\int_{\Gamma^{II}} \hat{p}_i \delta u_i^I \cdot z_i^{II} \, d\Gamma \]

The contact conditions for the normal and the tangential direction are different and are considered separately.

2.2.1 Normal Contact conditions

The conditions for normal contact are based on the physical requirement that two contacting bodies cannot penetrate each other, and hence, the normal traction is compressive. To be able to define the contact conditions mathematically, expressions for penetration and the contact tractions must be defined. Consider the distance between a fixed point on the surface \( S^I \) with coordinate vector \( x_i \) and an arbitrary point on the surface \( S^{II} \) (figure 2-2). Of particular interest is the minimum distance between \( x_i \) on \( S^I \) and \( S^{II} \). To find this minimum, one has to solve the problem.
\[ \|x_1 - x_2\| = \min \|x_1 - x_2^I\| \quad x_1 \in S^I, x_2 \in x_2^I \]

where \(x_2^I\) is representing all points on the surface \(S^I\). This minimum distance corresponds to the distance between \(x_1\) and the intersection point \(x_2\), i.e. the intersection of \(S^I\) with the line through \(x_1\), perpendicular to \(S^I\). Consequently, the minimal distance from \(x_1\) to surface \(S^I\) can be written as

\[ g(x_1) = (x_1 - x_2) \cdot n(x_2) \quad x_1 \in S^I, x_2 \in S^I \]  \[2.11\]

Where \(n(x_2)\) is the unit normal vector to \(S^I\) at \(x_2\) (as can be seen from figure 2-2). The function \(g\) is called the gap-function. The kinematical contact condition that bodies are unable to penetrate, is now formulated by \(g \geq 0\). The statement that the normal traction on the surface has to be compressive is the mechanical contact condition.

\[ \hat{p}_n = \hat{p}_3 \leq 0 \]

where \(\hat{p}_n\) is the component of the traction, normal to \(S^I\). With these definitions the conditions for frictionless contact can be written as

\[ \begin{align*}
\hat{p}_n \leq 0 \quad &\Rightarrow \quad g\hat{p}_n = 0 \\
g \geq 0
\end{align*} \]

The latter relation in this equation expresses that when the gap is positive, i.e. \(g > 0\), the normal traction \(\hat{p}_n\) becomes zero. On the other hand, when the bodies are contacting each other, the gap \(g\) is zero and the normal traction becomes compressive, i.e. \(\hat{p}_n \leq 0\).

### 2.2.2 Sliding and friction

Sliding and friction are movements and tractions in the plane tangential to the contacting surfaces respectively. Sliding of two contacting points is the difference in tangential displacement of the two contacting points. The discussion about friction is restricted to the representation of friction according to Coulomb. First sliding will be discussed, followed by a discussion about friction.

When two bodies are sliding relative to each other, a relative sliding velocity can be defined. To derive a formulation for the relative sliding velocity, the velocity of an arbitrary point \(x\) is required and defined as \(\dot{x} = \partial x / \partial t\). Likewise, the velocity vectors of the points \(x_1\) and \(x_2\) in figure 2-2 are indicated by \(\dot{u}_1\) and \(\dot{u}_2\). These velocities are decomposed in a normal and tangential component. For sliding the tangential velocity is required. The tangential velocity is determined by the inner product of the velocity vector and the tangential unit vector. This tangential unit vector, \(t = t_1z_1 + t_2z_2\), points in the direction of the tangential displacement. The relative sliding velocity is thus defined as

\[ v = (\dot{u}_2 - \dot{u}_1) \cdot t(x_2) \]
where \( v \) is the relative sliding velocity and \( t(x_2) \) is the corresponding unit tangential vector in point \( x_2 \). To move the hitting body relative to the target body, a resultant force is required which has the direction of \( t \). Therefore the sign of the resultant tangential force and the sign of the tangential velocity can be written as

\[
\text{sign} \ (v) = \text{sign} \ (p \cdot t)
\]

where \( p \) is the total force.

The resistance against sliding can be modelled by many different models, but the discussion is restricted to Coulomb’s friction. By that it is meant that contact is sticking for tangential tractions smaller than a certain maximum value and sliding for tractions that are larger than this value here. Moreover, the tangential traction remains constant when the maximum traction is exceeded and sliding occurs. Furthermore, the maximum tangential traction is related to the normal traction by the friction coefficient denoted by \( \mu \), which is independent of the velocity. The tangential traction \( p_t \) is measured by the components of the contact traction in the tangential direction \( p_1 \) and \( p_2 \). The conditions for friction are formulated by

\[
\hat{p}_t = \sqrt{\hat{p}_1^2 + \hat{p}_2^2} \\
\hat{p}_t \leq \mu \hat{p}_n
\]

The foregoing formulations for the tractions and displacements, in the normal and tangential directions, are formulated for continuous bodies and surfaces.

### 2.2.3 Discretization of the contact conditions

The previous considerations concerned continuous bodies and surfaces. As a result of the discretisation in FEM, and consequently, the discretisation of the contacting surfaces, the contact relations have to be written in terms of the nodal properties. It will turn out that this poses a problem for the integral equation of the virtual work that was discussed before. To avoid this problem, the virtual work is calculated for each node of the hitting body. After that, the contact of nodes and segments can be described. This is used to calculate the constraints for the nodal displacements. Many details about the discrete case can be found in Zongh (ref. (17)) and Kikuchi and Oden (ref. (5)).

Before considering the virtual work expression, it is useful to introduce some definitions for the contacting points and surfaces in the discrete case. Because a body is divided in elements, the surfaces are broken up in segments. These segments and adjacent nodes may get into contact with other bodies. All the nodes and segments on the surface of a hitting body are called hitting nodes and hitting segments. Even when they are not in contact. Similar terminology is introduced for the target body: the target segments and target nodes. In FEM-practice the deformation of the bodies and surfaces is given by their nodal displacements. To make it possible to calculate the intermediate displacements of a point on a segment, interpolation is required. The displacements found in that way will be used to calculate the virtual work. The isoparametric interpolation functions will be denoted by \( \phi \). They are used to interpolate the displacements of the adjacent nodes to get the displacement of a point with natural coordinates \((\xi, \eta)\). This penetration can be formulated by
\[ \phi_1 (\xi, \eta) = \frac{1}{4} (1 - \xi \lambda) (1 - \eta \lambda) \]
\[ \phi_2 (\xi, \eta) = \frac{1}{4} (1 + \xi \lambda) (1 - \eta \lambda) \]
\[ \phi_3 (\xi, \eta) = \frac{1}{4} (1 + \xi \lambda) (1 + \eta \lambda) \]
\[ \phi_4 (\xi, \eta) = \frac{1}{4} (1 - \xi \lambda) (1 + \eta \lambda) \]

*Figure 2-3 Natural coordinates in 3D, surface is shaded*

\[ \Delta u (\xi, \eta) = \psi^\omega (\xi, \eta) \Delta u_\omega \quad \omega = 1, \ldots, m \]  

Summation is implied over the index \( \omega \), and it corresponds to a summation over the adjacent nodes. Figure 2-3 demonstrates a four nodded segment with the accompanying linear interpolation functions. Consequently, the target surface is represented by piece wise continuous functions. When the piece wise continuous displacement functions are calculated for the hitting and the target surface they can be substituted in the virtual work relation, equation (2.9). The surface integral is this relation is defined over the target surface. When the target surface is discretised, it reduces to an integral over target segments. For the displacements of the target surface this integral is well defined but for the displacements of the hitting surface problems raise. The representation of the displacements is not continuous over the edges of the segments and consequently the integral in the form of (2.9) can not be computed. There is thus no interpolation possible to define the virtual displacement of \( u^1 \). To deal with this problem it is useful to introduce nodal contact forces instead of contact surface tractions. The virtual work is then calculated as the sum of the virtual work \( \delta a \) done by the nodal contact forces

\[ \delta A_c = \sum_{c=1}^{m} \delta a_c \]  

where \( c \) is the index for the hitting nodes and \( m \) the number of nodes that are in contact.
Figure 2-4 Demonstration of the hitting node and target segment

Contact between discretised contact bodies can be described in several different ways. Firstly, node to node contact can be defined. However, the usefulness of this approach is very limited because the meshes are, in general, not equally spaced and it will be hard to define which target node is contacted by the hitting node. Secondly, segment to segment contact, the type of contact discussed above, gives problems with the formulation. As from now on, the exposition will be for contact from node to segment. The relations (2.9) and (2.11) are defined for the target surface, and therefore, hitting nodes that contact target segments will be considered. This combination of a hitting node and a target segment will be denoted as a contact pair. Moreover, it is necessary to give equivalent nodal definitions for the positions, displacements and gaps of the hitting and the target bodies.

The contact conditions are considered for an arbitrary point on the hitting surface that contacts the target surface. The integral over the points on the hitting surface is changed in a loop over the hitting nodes. Therefore, $x_{kl}$ is substituted for $x_1$, where $k$ is the index for the number of the hitting node and $l$ the index for the hitting body. The target point $x_2$ is the point on the target surface with the minimum distance to the hitting point (figure 2-4). This point, on the target segment, is given by $x_2 = \phi^\omega x_{oll}$, where summation is applied over $\omega$ ($\omega = 1, 2, 3$ in the figure). The index $\omega II$ is for adjacent nodes of the target segment of body II. The nodes on the target surface are indicated with index $k II$ and the position of a node will be denoted by $x_{kII}$. The displacements according to the nodes with position $x_{kl}$ and $x_{kII}$ are $u_{kl}$ and $u_{kII}$, respectively. For the normal $n(x_2)$ the notation will be simplified to $n$, leaving the superscript II for body II. This normal is no longer continuous varying along the surface but will be constant for a segment and discontinuous over the edges of the target segment.

With use of equation (2.9), the virtual work $\delta w$ done by the nodal forces $q_i$ now is

$$\delta a_c = q_i \delta (\phi^\omega \Delta u_{oll} - \Delta u_{kI})^T n_i \quad \omega = 1, \ldots, m \quad i = 1, 2, 3 \quad 2.14$$

where summation is carried out over $\omega$ and $m$ is the number of adjacent nodes. The index $i$ is to indicate the direction with respect to the target segment, thus $i = 3$ is the normal direction and $i = 1, 2$ gives the tangential direction, equivalent to the definition of $z_i$, $i = 1, 2, 3$. The relation for the virtual work gives an additional force vector for the contact forces in the equilibrium equations.
Figure 2.5 Illustration of the discretized contact conditions

The next step is to discretize the contact conditions. The gap between the hitting point and the target surface for the continuous bodies, (2.11), is defined in local directions. The discretized version of this definition must be transformed to the global directions to make it compatible for the FEM computation. Hence, the gap for a hitting node, contacting a target segment, will be stated in the global directions. To be able to compute this gap in the global directions, the normal is decomposed in three components for the global directions. The normal vector in the global direction with index $i$ can be calculated by $e_i \cdot n_i = e_i$ and by this means the gap in the direction $i$ can be formulated by

$$ g_i(x_{kl}) = [x_{kl} - \phi \omega x_{oll}]^T n_i $$

where, again, summation is applied over $\omega$. Computation of the latter for $i = 1, 2, 3$ gives three components of the gap between the hitting node $x_{kl}$ and the target segment. In the incremental analysis an arbitrary step from time $t$ to time $t+\Delta t$ is assumed. The gap for the time $t$ and $t+\Delta t$ is denoted by $g_i^t$ and $g_i^{t+1}$, respectively. Accordingly, the gap $g_i^{t+1}$ can be computed from

$$ g_i^{t+1}(x_{kl}) = g_i^t(x_{kl}) + \Delta g_i^{t+1}(x_{kl}) $$

$$ = g_i^t(x_{kl}) + [n_j^{t+1}]^T [\Delta u_{kli}^{t+1} - \phi \omega \Delta u_{oll}^{t+1}] $$

$$ j = 1, 2, 3 $$. 

(2.15)

The complexity of the nonlinearity of the contact analysis can be deduced from this relation. To calculate the gap, the displacements and normals of the surfaces are needed but these are initially unknown. Taking small steps, and assuming that the surfaces are smooth allows one to replace the normal at $t + \Delta t$ by the normal at step $t$. Furthermore, the gaps are equal to zero when the hitting nodes contact the target segments, and therefore, the latter relation is the constraint for normal contact in the global directions. Carrying out equation (2.15) for all the hitting nodes gives matrix equation given by

$$ A^{t+1} \Delta u^{t+1} + g^i = 0 \quad g^i \in R_{j\eta} \quad A^{t+1} \in R_{j\eta} \times R_N $$

(2.16)

where $g^i$ is the vector of gaps (which has the dimension of three times the number of hitting nodes $n$). The matrix $A$ is the matrix of the global projection of the nodes of body I on the surface of body II for the configuration for time $t+\Delta t$. The dimension of the vector of the incremental displacements is $N$ which is the number of unknowns of the problem. The situation is illustrated in figure 2.5.
2.3 Imposition of constraints

There are several ways to introduce the contact conditions in the finite element method. These methods can be divided into two categories. In the first place, methods where all the degrees of freedom as well as the constraints are integrated in one formulation for the equilibrium under constraints. In the second place, methods where the number of degrees of freedom is reduced to a minimum using the constraint equations. Of the first category, two widely used methods, the Lagrange multiplier and the penalty method are considered briefly. Of the second category one method will be considered. The latter method, the so-called tying method, is considered in more detail because this method is not commonly used and therefore, not easily found in the literature. This tying method will be used for the discussion of the incremental procedure in chapter 4. The Lagrange and penalty method are considered just to point out several differences and to demonstrate that the problems that exist for the tying method, also exist for the other methods. A lot of details can be found in Zongh (ref. 17). It will be assumed that a search algorithm finds the contacting pairs but this search algorithm will be discussed later.

2.3.1 Lagrange multiplier method

With the Lagrange multiplier method, the virtual work expression (2.5) is augmented with the Kuhn-Tucker condition, \((\Lambda A u + g)^T \Lambda = 0\), introducing a new set of \(m\) variable(s), the Lagrange multipliers. This augmented form of the virtual work is in this case:

\[
\delta A^* = \delta u^T f(u, \lambda) + \delta u^T A^T \Lambda + (\Lambda A u + g)^T \delta \Lambda = 0
\]

where \(\Lambda\) is the vector of the Lagrange multipliers. This modification leads to the set of \(N+m\) dimensional equations

\[
\begin{align*}
    f(u, \lambda) + A^T \Lambda &= 0 \\
    \Lambda A u + g
\end{align*}
\]

When it is necessary to consider the linearised version of these equations, they read

\[
\begin{bmatrix}
  K & A^T \\
  A & 0
\end{bmatrix}
\begin{bmatrix}
  \Delta u \\
  \Delta \Lambda
\end{bmatrix}
+ \begin{bmatrix}
  \Delta \lambda \\
  g
\end{bmatrix}
= 0
\]

The method increases the number of unknowns but yields the exact solution. It is noted that in (2.19) the diagonal elements in the coefficient matrix corresponding to the Lagrange multipliers are zero. It is therefore expedient to rearrange the system of equations as presented in (2.19). But also in that case, the matrix can become ill-conditioned or singular. A real disadvantage of the method is, that the vector of incremental changes contains displacements and forces, which has a negative effect on the formulation of the path-following method that is discussed in the next chapter. On the other hand, an advantage is that with the multipliers the displacements and the contact forces are automatically known.

2.3.2 Penalty method

With the penalty method the contact conditions are also introduced in the relation of the virtual work. In this case, the virtual work (2.5) is augmented to
\[ \delta A = \delta u^T f(u, \lambda) + \alpha (A\Delta u + g)^T \delta (A\Delta u + g) = 0 \]  

2.20

The latter term in the relation corresponds to the virtual work associated with an elastic spring with stiffness \( \alpha \) which resists the motion in accordance with the condition (2.16). The method therefore approximates the contact conditions for finite values of \( \alpha \). The exact solution is obtained when \( \alpha \) goes to infinity.

The relation (2.20) must hold for an arbitrary variation of the displacements so that

\[ f(u, \lambda) + \alpha A^T (A\Delta u + g) = 0 \]

The linearisation of these equations reads

\[ -(K + \alpha A^T A) \Delta u = I\Delta \lambda + \alpha A^T g \]  

2.21

Hence, using this technique, a large value is added to the diagonal elements of the stiffness matrix and a corresponding force is added to the force vector so that the contact condition is fulfilled. The choice of the penalty \( \alpha \) can be a problem when the structure has a vastly varying stiffness distribution. Moreover, the method yields an approximated solution which depends on the penalty number. On the one hand, when the penalty number is not properly chosen, e.g. when it is too small, it can result in unacceptable penetrations. On the other hand, a large penalty number reaches the exact solution but can cause an ill-conditioned system to solve. The penalty numbers follow not automatically from the problem definition and, thus, should be defined by the user. Several relations for the optimal penalty parameter are introduced in the literature. An advantage of the method is, however, that no additional number of unknowns are needed.

2.3.3 Tying method

In the introduction the so-called tying method is described as a method where the degrees of freedom are reduced to a minimum, making use of the constraint equations. The reduction of unknowns is accomplished by a coupling of the hitting nodes and the nodes adjacent to the target segment. This coupling is called a tying, and a detailed description of the tyings is discussed later. The displacements, normal to the target segment of a contact pair can be coupled. Accordingly, the displacements of the hitting node are tied to the displacements of the target segment. In case of frictional contact, the tangential displacements can be tied as well. A disadvantage of this method is that when a converged solution is found, with or without the tangential tyings, an additional loop is required to search for the friction forces that exceed the friction limit. The nodes that exceed this limit should be released or tied. When these nodes are released or tied, the configuration is not in equilibrium in the tangential direction. Therefore, an additional iteration is required to solve the tangential constraints. The other two methods introduce the constraints in the formulation and, thus, such an additional loop is not needed. For the tying method nodes are tied when they are penetrating, and consequently, the search algorithm is more critical than for the other two methods. Furthermore, the tying method does not influence the variational formulation. For this method, a distinction must be made between deformable and rigid contact. A rigid body is undeformable and, therefore, not divided in elements. Because no tyings can be created, a system of local boundaries is defined.

25
In equation (2.12) the displacements of a point on the segment were found by interpolation of the nodal displacements of the adjacent nodes. With this relation, the displacement vector of a hitting node \( k \) can be written as a function of the displacement vector of target point on the segment. This relation, the so called tying relation is

\[
\Delta u_{k1}(\xi, \eta) = \phi^\omega(\xi, \eta) \Delta u_{\omega} \quad \omega = 1, \ldots, m
\]

where \( m \) is the number of adjacent nodes of the target segment. Notice that this relation is defined in the local directions. The tying relation in the global directions is discussed later, but it turns out to be still of the form

\[
\Delta u = N^\omega \Delta u_{\omega}
\]

Carrying out all the tyings will result in a reduced displacement vector \( \Delta \bar{u} \) and a tying matrix \( S \) defined by

\[
\Delta u = \Delta (S\bar{u}) = S\Delta \bar{u} \quad \Rightarrow \quad u \in \mathbb{R}^N, \quad \bar{u} \in \mathbb{R}^{N-n}
\]

where \( \Delta u \) and \( \Delta \bar{u} \) are in the global directions. The tying matrix \( S \) has the dimensions \((N \times N-n)\) with \( N \) the number of unknowns and \( n \) the number of removed or tied unknowns. Making use of the linearization of \( f(u, \lambda) = 0, K\Delta u + \lambda \Delta \lambda = 0 \), and substituting the latter relation gives

\[
-KS\Delta \bar{u} = \Delta \lambda I
\]

Pre-multiplying this relation by the transposed tying matrix \( S^T \) gives

\[
-KS\Delta \bar{u} = \Delta \lambda I
\]

where, \( \bar{K} = S^T KS, I = S^T I \) are the reduced stiffness matrix and a reduced load vector, respectively. In general the reduced vectors can be found by pre-multiplying with the transposed tying matrix \( S^T \)

Accordingly, the contact constraints are used to reduce the system of equations from \( N \) to \( N-k \) degrees of freedom.

The constraints in the direction tangential to the surface can be introduced by the tyings as well. Consequently, the hitting nodes are tied in all the directions, normal and tangential. By that means, the contact is assumed sticking. Nevertheless friction is introduced with a loop over the hitting nodes to assure that, for each node, the maximum friction force is not exceeded. When the maximum friction force is exceeded the node can be released and a new solution will be found. Next, when the friction force for sliding nodes is less than the maximum, the node must be tied again. Numerically it is preferable to have a continuous function for the friction force, instead of sticking contact with a limiting value for the friction force. Accordingly, the tying can be made flexible. Then the hitting nodes are tied with a non-linear spring. The stiffness characteristics of the spring depends on the friction law that is used. For
Coulomb's friction, the friction force can be expressed as a function of the relative sliding velocity and the normal force. This function is an approximation to the discontinuous contact condition and the smoother the function is made the less natural the friction is taken in account. On the other hand an accurate approximation gives numerical problems because of flipping between the positive and negative direction of the friction force.

The displacements of the surface of a rigid target body are related to one representative point of the body. As mentioned before, this is possible because the geometry of the body does not change. Because this rigidity of the body, it is not discretized, and the hitting nodes of the deformable body can not be tied to the target segments. Therefore the displacements of the hitting nodes should be subscribed by the displacements of the rigid body, normal to the rigid surface (figure 2-6). Recalling the linearization of \( f(u, \lambda) = 0, K \Delta u + \lambda \Delta \lambda = 0 \), there are two possibilities to fulfill the constraints for the displacements. At first, it is possible to replace the unknown displacements \( \Delta u \) by the displacements following from the constraints, but it is numerically preferable to use a second option. This second possibility is to add a large value to the diagonal term of the stiffness matrix, and add a corresponding term to the force vector, such that the constraint is fulfilled. This can be formulated by

\[
- \left[ K + \sum_{n=1}^{m} \alpha e_n e_n^T \right] \Delta u = \Delta \lambda I + \alpha \Delta u_c \quad \Delta u_c \in \mathbb{R}_N
\]

2.25

where \( \alpha \) is a large value, \( \alpha >> \max(k_{ij}) \) and \( e_n \) is the unit base vector in the direction of the n-th degree of freedom. The vector \( \Delta u_c \) contains zeros for the terms corresponding to the unconstrained displacements and non-zeros for the constrained displacements.

2.3.4 Solving the tyings

The subject of the following is the tying and the transformation from the local defined tying to the tying in the global directions. These subjects are introduced in two demonstrations. The tyings in the tangential direction are left aside because, in case of sticking contact, there are no problems with transformations from the local directions to the global directions.
The first demonstration is a tying in 2D as plotted in figure 2-7. Although it should be mentioned that the relations for 3D are similar. The difference between 2D and 3D is the number of target nodes and the number of interpolations, which is required for two directions in the direction tangent to the target surface. In this demonstration, hitting node A is tied to the target nodes B and C, figure 2-7. Furthermore, it is assumed that the global and local directions are parallel. The tying relation in the normal direction \( \hat{u} \) is formulated by

\[
\Delta \hat{u}_A = \frac{1}{2} (1 - \xi_A) \Delta \hat{u}_B + \frac{1}{2} (1 + \xi_A) \Delta \hat{u}_C
\]

where \( \hat{u} \) is the displacement normal to segment BC, and \( \xi \) is the natural coordinate along this segment. When the global and local displacements are similar, the matrix equation yields

\[
\Delta \mathbf{u} = S \Delta \hat{u} \quad \Leftrightarrow \quad \mathbf{u} = [u_A, u_B, u_C]^T \quad \hat{u} = [u_B, u_C]^T
\]

\[
S = \begin{bmatrix}
S_{AA} & S_{AB} & S_{AC} \\
S_{BA} & S_{BB} & S_{BC} \\
S_{CA} & S_{CB} & S_{CC}
\end{bmatrix} = \begin{bmatrix}
\frac{1}{2} (1 - \xi_A) & \frac{1}{2} (1 + \xi_A) & 0 \\
0 & 1 & 0 \\
0 & 0 & 1
\end{bmatrix}
\]

The latter can be generalized for a node \( k \) tied to the nodes \( l \) and \( m \). The relation then will be

\[
\Delta \mathbf{u} = S \Delta \hat{u} \quad \mathbf{u} = [u_1, u_2, \ldots, u_N]^T
\]

\[
\hat{u} = [u_1, \ldots, u_{k-1}, u_{k+1}, \ldots, u_l, \ldots, u_m, \ldots, u_N]^T
\]

Where the \( k \)-th row of the tying matrix is the tying relation between the three nodes \( k, l, m \). All the terms except of the terms with equal index (which are 1) and the terms \( S_{kl} \) and \( S_{km} \) in the tying matrix are equal to zero. Notice, that the dimensions of the tying matrix are \( N \times (N - n) \), where \( n \) the number of tyings.

In a second example, a transformation will be introduced to transform one of the local directions to the global directions, such that the displacement, normal to the target surface is prescribed. The tying is not simply transformed to the global direction because in that case one is unable to remove a degree of freedom. With figure 2-7 it is possible to define both, a tying relation and a transformation of the local directions to the global direction, by

\[
\Delta \hat{u}_A = \phi^\omega (\xi_A) \Delta u_\omega \quad \Rightarrow \quad \omega = B, C
\]

\[
\Delta \hat{u} = \cos \beta \cdot \Delta u_1 + \sin \beta \cdot \Delta u_2
\]

where

\[
\phi^B (\xi_A) = \frac{1}{2} (1 - \xi_A), \quad \phi^C (\xi_A) = \frac{1}{2} (1 + \xi_A)
\]

2.26
Figure 2-7 Tying where the normal of the target is not in the global directions

These relations for the normal displacement $\bar{u}$ of the hitting node have to be satisfied simultaneously. Substitution of the second relation of equation (2.26) in the first relation of equation (2.26) and working this out gives an equation in the two global directions. One of these global directions should be isolated. Which direction is chosen can be based on the angle $\beta$, i.e., by choosing the global direction which is the closest to the normal. This choice is equivalent to the selection of the coordinate for which $\cos \beta$ or $\sin \beta$ in equation (2.26) is the largest. Assume e.g. that the $\cos \beta$ is the largest, and thus $u_1$ will be isolated, the tying relation for point A in figure 2-7 then yields

\[
\Delta u_1 (A) = \frac{1}{2} (1 - \xi_A) \{ \Delta u_1 (B) + \zeta \Delta u_2 (B) \} + \\
+ \frac{1}{2} (1 + \xi_A) \{ \Delta u_1 (C) + \zeta \Delta u_2 (C) \} - \zeta \Delta u_2 (A)
\]

\[
\zeta = \frac{\sin \beta}{\cos \beta}
\]

Next, the other direction $\Delta u_2 (A)$ is computed and when this is substituted in (2.27) it gives a displacement in the first direction corresponding to the tangential displacement of point A. The relations for the cases that, firstly, $\sin \beta$ is larger, and secondly, the tying is in 3D, can be found by a similar approach. The latter shows clearly that, with the transformed tyings, the displacement in the first global direction is a function of the second global direction of the hitting node. Now, return to equation (2.22) for the displacement of a hitting node $\Delta u_{k1}$

\[
\Delta u_{k1} (\xi, \eta) = \phi^0 (\xi, \eta) \Delta u_{\omega 11} \quad \omega = 1, \ldots, m
\]

When this tying relation is compared with equation (2.27), it can be seen that at the right hand side an additional term is required for two other global displacements of the hitting node. The displacement for the tied direction $\Delta u_\omega$, of a hitting node in 3D, can in general be written as

\[
\Delta u_\omega (\xi, \eta, \beta) = N^\omega (\xi, \eta, \beta) \Delta u_\omega \quad \omega = 1, \ldots, m \quad i = 1, 2
\]
where $\xi$ and $\eta$ are the natural coordinates and $\beta_i$ is the angle of the normal $n$ with the global directions with index $i$. Consequently, the functions $N^{0}$ relate one of the displacements of the hitting node to firstly, the displacements of the adjacent nodes, and secondly, the remaining two directions of the hitting node, all in the global directions. This relation is such that the normal displacement of the hitting node is equal to the normal displacement of the target segment. The number of $m$ is therefore the number of the adjacent nodes of the target segment times the three directions plus the two global directions of the hitting node closest to the tangential direction. With equation (2.28) the tying matrix can be formed.

2.4 Contact search

To determine the gap between the hitting node and the target segment, the displacements of the nodes that are in contact are required. With the nodal displacements one is enable to find the normals and, consequently, the contacting pairs. To find the contacting nodes, a contact search algorithm is needed. About contact search a lot has been written, the details and a general theoretical approach can be found e.g. Zongh (ref. 17) and in several other references. One subject which can be classified as a problem of the contact search, and of importance for the remaining discussion, is the problem of using tolerances.

For all the considered methods the contacting pairs, and thus which hitting node is (potentially) contacting which target segment, must be determined. For the different methods, the properties which are required to solve the contact constraints successfully are different. For the Lagrange and penalty method the gap is used to constrain the displacements for contact. The contacting pairs always have to be determined, even when no contact occurs. The Lagrange and penalty methods can anyhow be used when node to node contact is defined. In that case, the nodes that will contact each other must be known beforehand which is only possible when there is no sliding between the contacting bodies. The discussion is, from now on, focused on the tying method.

The input of this method are the penetrating nodes and the equivalent penetration of the target body. Notice that when the nodes are found to be in contact and tied to the target segment the gap is not required for that nodes, as long as they are in contact. On the other hand, for the nodes in contact a search for tension forces is required to determine the nodes that have to be released.

It is important to ensure that the formulation does not release nodes that are in contact. Therefore a tolerance on the surface is defined. This tolerance is defined such that a node, within a certain distance from the target surface, is assumed to be in contact. In this way, the chance that nodes are lost in the search algorithm becomes smaller. A drawback is the less natural modelling of contact when the tolerance is relatively large. Against this disadvantage stands an advantage: A larger tolerance on the target surface turns out to remove the necessity to split the increments. Splitting of an increment is generally necessary to deal with discontinuity in the process when nodes are contacting and are tied. This discontinuity and the accompanying splitting of an increment is subject at an other place. In figure 2-8 it is shown that when the surface near the contacting point is flat and very finely meshed, the number of nodes for which a splitting is needed, slows the calculation significantly. Hence, a tolerance on the target surface prevents splitting and speeds up the calculations. Difficulty with tolerances is the overlap of tolerances near corners because it is not defined in such an overlap (the target surface segment for the hitting node). In general, the problems where tolerances are required, can be seen as problems due to the nonlinearities in the calculation of problems with contact. For future reference, it is mentioned that all the problems for which tolerances are needed discussed so far, are at the element level and therefore a tolerance referred to the measures of the element will be sufficient.
Contact in FEM

Figure 2-8 Tolerance to speed up the calculation

![Diagram showing tolerance and hitting body](image)

nodes within tolerance will be assumed to be in contact

Figure 2-9 Displacement due to the tolerance

When a node is within the tolerance of the target surface it is assumed to be in contact and therefore it is tied to the target segment. Moreover, a displacement is needed to force the node to the surface as can be seen in figure 2-9. By this means, tolerances produce non-homogeneous terms in the tying relations. With equation (2.22) the local normal displacement of a hitting node positioned in a distance of the target segment, defined by the tolerance, is written by

\[
\Delta u_{k1} (\xi, \eta) = \phi^\circ (\xi, \eta) \Delta u_{oH} + \Delta u_{NH} \quad 2.29
\]

where \( \Delta u_{NH} \) is the vector of the non-homogeneous displacements. In the global direction of the tying, the relation transforms with equation (2.28) to

\[
\Delta u_T (\xi, \eta, \beta_1) = N^\circ (\xi, \eta, \beta_1) \Delta u_o + \Delta u_{NH} \quad 2.30
\]

where \( \Delta u_{NH} \) is the component of the vector of non-homogeneous terms in the tied direction. Notice that the displacements in the two other directions have to be added to the displacement vector. This problem of non-homogeneous terms is mentioned because it causes problems in the solution of the system of equilibrium equations by the incremental procedure.

The splitting of an increment is never exact so that in geometrically non-linear analysis the scaled positions of the hitting node are not precisely on the target surface. When the displacement of the hitting node is scaled too much, the position of the node is on a distance from the target surface. Consequently, when a solution is sought for this situation it will result in tension forces and the node will be released. This action slows the algorithm and flipping between contact and no contact can occur. In that case,
allowing a certain amount of tension forces to occur, can speed up the calculation. This problem has much in common with the problem of the previously mentioned contact tolerance, where tension forces are introduced when the nodes are forced to stick to the surface. Problem with a separation force is the choice of its magnitude, because the occurring forces are not known beforehand. Furthermore, the solution of the problem can change due to the separation force e.g. in problems with sliding.

2.5 Aspects of contact analysis

After the discussion of the previous possibilities of modelling mechanical contact, it is necessary to discuss some general aspects of mechanical contact in FEM analysis that have influence on the implementation of the incremental solution procedure. In this chapter we showed that the imposition of constraints changes the system of equations. The main problem for incremental iterative solution procedures in contact analysis is the discontinuity that contact imposes on the direction in which the solution proceeds. Besides tolerances are introduced which may cause problems for the solution procedure. These two subjects will be considered.

The first subject is discontinuity of the direction of the solution increment at contact. In general, before a hitting body contacts the target body, the stiffness of the construction is less than the stiffness after contacting. This results in smaller displacements and bigger steps for the external applied load. In other words, the behaviour of the configuration changes abruptly when the bodies are contacting. Moreover, when the bodies are hitting, degrees of freedom are introduced or removed so that the system of equations and the stiffness matrix are changing. The change of the stiffness matrix is due to the increase or decrease of unknowns for the Lagrange and the tying method, respectively. In case of the penalty method, the addition of terms to the stiffness matrix changes the stiffness matrix. Lagrange's method, for example, introduces the contact forces as multipliers in the vector of unknowns. The tying method removes unknowns by tying the normal degree of freedom of the hitting nodes to the target nodes.

Until now, it is assumed that the inertia forces can be neglected, in other words, it is assumed that the problems can be solved by a quasi-static solution procedure. Accordingly, problems occur in the quasi-static analysis when the inertia forces cannot be neglected any more and thus the problem is dynamic. This occurs e.g. when two bodies separate and large residual forces occur to bring the bodies to their equilibrium. Notice that this can occur locally too, due to high separation forces or due to the geometry, when only a few nodes are released and, consequently, only a part of the structure contains high residual forces. The same problem of residual forces occurs when tolerances are defined to speed up the calculations. Hence, both the introduction of a large tolerance on the target surface, and the introduction of separation forces can cause problems.
Chapter 3

Path-following methods

3.1 Incremental approach

With the governing equations for the displacements, \( u \in \mathbb{R}_N \), and the load parameter \( \lambda \), as derived in the previous chapter, a vector \((u, \lambda)\) has to be determined from a set of nonlinear equations

\[
f(u, \lambda) = 0, \quad f : \mathbb{R}_{N+1} \rightarrow \mathbb{R}_N
\]

where \( \lambda \) and \( u \) are defined in such a way that \((u, \lambda) = (0, 0)\) is the undeformed equilibrium state. The set of solutions \{\((u, \lambda)\)\} that satisfies (3.1) is therefore a one dimensional family of solutions in \( \mathbb{R}_{N+1} \). This set of solutions of (3.1) represents a curve \( C \) or a set of curves in \( N+1 \) dimensional space spanned by \( u \) and \( \lambda \). A parametric representation of the curve \( C \) can be defined by introducing a so called path parameter \( \eta \). The solutions \((u, \lambda)\) of the system (3.1) can therefore be written as \((u(\eta), \lambda(\eta))\).

The system (3.1) consists of \( N \) equations and \( N+1 \) unknowns. This system is therefore underdetermined and one of the unknowns must be prescribed to reduce the \( N+1 \) degrees of freedom to \( N \). In that case, one is able to compute one or more isolated solutions \((u^*, \lambda^*)\) of the system in equation (3.1). Isolated solutions \((u^*, \lambda^*)\) are solutions with the property that in the direct neighbourhood of that solution no other solutions exist. When the system (3.1) has more than one isolated solution, the solutions can still be determined unless these solutions are multiple. In the general case, equations (3.1) admit multiple but isolated solutions and a procedure to obtain these is discussed in the following.

Instead of reducing the number of degrees of freedom from \( N+1 \) to \( N \), one is able to specify an equation that is independent of the system (3.1), and defines the path parameter \( \eta \). When this extra equation \( f_{N+1} \) is added to the system (3.1), an augmented system of \( N+1 \) equations results. These \( N+1 \) equations, \( f_1, f_2, ..., f_{N+1} \), have \( N+1 \) unknowns, \( u_1, u_2, ..., u_N \) and \( \lambda \), for a prescribed value of the path parameter \( \eta \). With the use of \( \psi^T = (u^T, \lambda) \), this system can be formulated by

\[
F(v, \eta) = \begin{bmatrix} f(v) \\ f_{N+1}(v, \eta) \end{bmatrix} = 0 \quad F : \mathbb{R}_{N+1} \rightarrow \mathbb{R}_{N+1}
\]

which is determined, so that, when solutions exists, it has one or more isolated solutions \((u^*, \lambda^*)\) for a prescribed value of the path parameter \( \eta \). The path parameter can be defined in an infinite number of ways. One of these is the load parameter \( \lambda \), which represents a means to vary the intensity of the externally applied load. This choice results in an extra equation \( f_{N+1} = \eta - \eta_k = \lambda - \lambda_k = 0 \), where \( \lambda_k \) is a prescribed value of the load. The problem then is to find the solution \((u, \lambda = \lambda_k)\). Alternatively, a displacement component can be chosen. This gives for the path parameter \( f_{N+1} = \eta - \eta_k = u_\alpha - u_{\alpha k} = 0 \), where \( \alpha \) is the number of an arbitrary nodal displacement component, and \( u_{\alpha k} \) the displacement of node \( \alpha \) in step \( k \).
These added equations describe surfaces for the chosen parameter. For obvious reasons, these surfaces are only appropriate if they are not parallel to the curve C, in the region of interest. When the solution curve and a surface are nearly parallel, and the surface is slightly moved in the direction normal to the surface, the intersection of both will move over a large distance. In that case, the configuration changes significantly for small perturbations of the path parameter, and this situation leads to an ill-conditioned system of equations. Accordingly, if a small change of the selected path parameter $\Delta \eta$ results in significant changes for the degrees of freedom $\Delta v$, ill conditioning of the system occurs. An ill-conditioned system of equations causes problems for the stability of the numerical computation. An extra equation $f_{N+1}$, which avoids the occasional occurrence of this singular condition was proposed by Riks [10] in 1970. This is a choice of $\eta$ which is continuously adapted along the path. Details of the definition of this surface $f_{N+1}$ are discussed later.

The isolated solutions $v^*$ of the system (3.2) are the zeros of the nonlinear function $F(v, \eta)$. These solutions can be found numerically by using an iterative method. A useful iterative process generates an infinite sequence $v^i$, $i = 1, 2, 3, \ldots$, for an initial value $v^0$, that converges to an isolated solution $v^*$. The superscript $i$ is the index for the $i^{th}$ cycle of the iterative process and the corresponding solution $v^i$ is called iterate $i$. Notice that $v^i = (u^i, \lambda^i), i = 0, 1, 2, \ldots$ is always an approximation to the exact solution $v^* = (u^*, \lambda^*)$. When a process is converging it means that

$$\lim_{i \to \infty} v^i = v^*$$

From now we will adopt the convention that when an iterate $v^i$ is ‘close enough’ to the isolated solution $v^*$, and is accepted as a solution, it will be denoted with $v$. To make it possible to decide whether an iterate is ‘close enough’ to the isolated solution $v^*$, convergence criteria are required, so that the iteration process can be stopped.

With a given iterative method, the set $E$ of all initial values $\{v^0\}$ that converge to $v^*$ can be small or big, dependent of the degree of the nonlinearity of the problem. With regard to the convergence behaviour, problems can be divided in two classes. Firstly, when the set $E$ is the entire domain of $v$, problems have global convergence behaviour, and secondly, when the set $E$ contains only the neighborhood of a solution $v^*$, problems have local convergence behaviour. In the case of local convergence, an initial value $v^0 \in E$ causes the iterative process to diverge and, consequently, the process does not result in the solution $v^*$. To prevent this process from making useless and expensive calculations, it is necessary to use a criterion that decides when the process is diverging so that it can be stopped. Consequently, termination criteria are necessary to control the calculation.

We now return to the problem of two or more isolated solutions for a fixed value of the path parameter. As mentioned, this is a general problem in nonlinear analysis. Let the system of equation (3.2) have more than one isolated solution for a fixed value of the path parameter $\eta_k$ that can be found numerically, say $(v_k^*)_1, (v_k^*)_2, (v_k^*)_3, \ldots$. The solution $(v_k^*)_2$ is a solution of the equilibrium system (3.2), however, when this solution is taken as a guess for the computation of $(v_k^*)_1$, it will never converge to $(v_k^*)_1$ because $(v_k^*)_2$ is not in the domain of attraction of $(v_k^*)_1$ but in the domain of attraction of itself. Consequently, nonlinear problems exhibit in general a local convergence behaviour, and the initial value for the iterative process must be near the solution $v_k^*$. The iterative process is now characterized as a procedure to compute one solution out of the set of multiple isolated solutions $(v_k^*)$. Which solution will be found, depends on the initial value $v^0$. 

34
Because of this local convergence behaviour, it is in general not possible to compute an arbitrary solution of the curve \( C \) when the undeformed state \((0,0)\) as the initial value of the iterative process. A better procedure to find such a solution is to follow the curve \( C \) in a stepwise manner, until the required value of e.g. the load or the deformation is reached. Accordingly, the global solution of the system (3.2), the curve \( C \) in the \( N+1 \) dimensional space, can be found by the computation of a sequence of solutions \( v_k^*, k = 1,2,3,... \) for increasing values of the path parameter \( \eta_k \). This sequence of solutions thus represents the curve \( C \) in a point wise manner. Each solution of the sequence \( v_k, k = 1,2,3,... \) can be calculated by an iterative process \( v^i_k, i = 1,2,3,... \). Therefore, the process to compute the curve \( C \) consists of successive prescription of a new step \( \Delta \eta_{k+1} \rightarrow \eta_{k+1} = \Delta \eta_{k+1} \), and for each step the calculation of a guess \( v^0_{k+1} \), as the initial value for the iterative process, followed by a sequence of iterations to find the solution \( v_{k+1} \). The methods based on this process are often called incremental-iterative methods or path-following methods. As follows from the previous discussion, they make use of a predictor-corrector scheme. Figure 3-1a illustrates this concept. The predictor for step \( k+1 \) computes a guess \( v^0_{k+1} \) by using the solutions of the latter and the previous computed solutions \( v_k, v_{k-1}, v_{k-2}, ... \). Notice, that the simplest prediction is the use of the value of the latter solution \( v_k \). The correction is a number of iterations to find the solution \( v_{k+1} \), starting from the initial value \( v^0_{k+1} \).

If the solutions \( v_k, k = 1,2,3,... \) are interpolated to get an approximation of curve \( C \), it is important that the solutions are regularly spaced over the curve \( C \) (figure 3-1b). This clearly points out a difficulty for the step algorithm, and it turns out that, the effectiveness of the stepping algorithm depends crucially on two choices that must be made:

- the choice of the direction into which a next step should be made (but when the parametrization is well defined this is fulfilled automatically)
- the size of the step along this new direction.

Before continuing the discussion of the path following method it is useful to discuss a number of simple examples. As was pointed out by Riks in [10] infinitely many choices for the path parameter are possible and the first choice that is discussed is the load parameter as path parameter, \( f_{N+1} = \eta - \eta_k = \lambda - \lambda_k = 0 \). The method of this load stepping prescribes the load and calculates the corresponding displacements. In all the following methods, the undeformed equilibrium state \((u,\lambda) = (0,0)\) can be used as the starting point for the calculation because it is a solution of the system (3.2). For the method of load stepping a subsequent isolated solution on the curve \( C \) can be found when the load parameter is changed to \( \Delta \lambda \) (figure 3-2a). For the first step, \( \lambda_1 = \Delta \lambda \), the solution \( u(\Delta \lambda) \) must be found that satisfies the
equilibrium of equation (3.2) i.e. the displacements \( u_1 \) corresponding to \( \lambda_1 \). When \( v^0_1 = (0, \lambda_1) \) is used as initial value (or prediction), and \( v^0_1 \) is substituted in (3.2), \( F(v^0_1) \) is not in equilibrium and gives a residual force vector \( r \), by \( F(v^0_1) = r^0 \). The superscript in \( r^0 \) indicates the residual force vector after the prediction. This residual force vector can be seen as the out of balance forces that are needed to keep the construction in inequilibrium at \( v^0_1 \). Next, the iterative process aims to reduce the residuals for the fixed value of the load \( \lambda_1 \), with the sequence \( v^i_1 \), \( i = 1, 2, 3, \ldots \). When the prediction \( v^0_1 \) is in the domain of attraction for the solution \((u_1, \lambda_1)\), and the iterations are repeated over and over again, ultimately, the residuals eventually will vanish. Each cycle therefore improves the last computed iterate \( v^i_1 \) by applying an iteration function to \( v^i_1 \) such that an improved iterate \( v^{i+1}_1 \) is obtained. However, the “exact” solution can never be found and therefore a termination criterion is used to stop the iterative process. When the iterative process is terminated, the last computed iterate \( v^{i+1}_1 \) is accepted as the solution \( v_1 \). Once the solution for the first step is obtained, the same procedure can be used to calculate the solution \((u, \lambda_2 = \lambda_1 + \Delta \lambda)\) for the next step and so on.

The load stepping procedure is simple and straightforward. However, it cannot be applied to a class of problems which are quite common in geometrical nonlinear analysis i.e. problems that have a force history as showed in figure 3-2b. This phenomenon is characterized by a part of the curve where, for ongoing deformation, the load must be decreased. This part of the curve is marked by point A and D. Point A is a limit point for the load because the load has a (local) maximum value \( \lambda_c \). It is at this point where the stability of the calculations will be lost because of the ill-conditioned state of the system. Furthermore, when at point A in the figure the load is increased, the iterative process most likely diverges. In some cases it may converge to point B. This behaviour of the incremental process is unacceptable when parts of the solution curve beyond the limit point at \( \lambda_c \) are required. The intermediate part of the solution curve C, between the points A and B, is then not defined by the chosen parametrization. When, however, in the demonstration of figure 3-2b the displacement component \( u_\alpha \) is chosen as path parameter, the total curve C can be described by the path parameter. The corresponding added equation is \( f_{n+1} = \eta - \eta_k = u_\alpha - u_{\alpha k} = 0 \), and this method is referred to as the method of prescribed displacements. The solutions for prescribed loadsteps, in figure 3-2b, are indicated with dots. The solutions for prescribed displacements are indicated by asterisks. Figure 3-3a gives an equivalent problem of a limit point for the displacement component \( u_\alpha \). Problems with turning points for the load are called snap through problems.
Turning points (or limit points) are singular points for the system of (3.2). This singularity is due to the definition of the system, i.e. by the choice of the added equation $f_{N+1}$. As mentioned before, this singularity can be solved by introduction of an other, more appropriate, auxiliary surface. The latter is introduced in such a manner that for each point of the curve $C$, an intersection of this curve with the auxiliary surface is well defined. This guarantees the regularity of the resulting system of equations.

It should be noted that a second class of singular points that can occur, the branching or bifurcation points, cannot be solved by changing the extra equation $f_{N+1}$. A branching point remains a branching point and thus represents a singularity for each choice of the auxiliary surface. The branching points can be solved by additional measures but these are out of the scope of this report, see Seydel [15] and Riks [10].

In general, a solution of the system in equation (3.2) satisfies both the system $f(y) = 0$ and the added equation $f_{N+1} = 0$. When an auxiliary surface is used that is continuously adapted along the path, it can be viewed as control surface for the iterative process. Accordingly, it assures that when the initial value is chosen in the domain of attraction $E$, the iterative process is stable and ultimately meets the isolated solution $y^*$ (see figure 3-1a). Moreover, when the auxiliary surface is properly chosen it assures a well defined parametrization of the curve $C$ as can be seen in figure 3-3b.

Before resuming the process of the path following method, it is useful to consider the efficiency and the accuracy of the method. These two properties of the method are conflicting and therefore a trade-off between the two has to be made. The efficiency and the accuracy are, among others, influenced by the steplength. Consider therefore the path-following process of stepping along the curve $C$. On the one hand, when the steps are too large (assuming that the process is converging which is not certain) the process is efficient but, the representation of the curve is poor and details can be lost. On the other hand when the steps are too small, the representation of the curve $C$ is excellent but the computational effort may be unnecessarily high. To determine the step length, and to be able to change it during the analysis, a measure for the length between two points along the curve $C$ is required. The length of a segment between two points on the solution curve $v(\eta_k)$ and $v(\eta_{k+1})$, the so called arclength, can be defined as the line-integral along the solution curve from point $\eta_k$ to $\eta_{k+1}$.
\[ S_E = \int_{\eta_k}^{\eta_{k+1}} ds = \int_{\eta_k}^{\eta_{k+1}} \sqrt{\left( \frac{dv}{d\eta} \right)^2 + \left( \frac{dv}{d\eta} \right)^2} d\eta \]

Because this arclength is hard to compute in the FEM practice, a so called pseudo arclength is defined. This pseudo arclength is the Euclidian norm of the differential vector, connecting two neighbouring points \( v(\eta_{k+1}) - v(\eta_k) \). This can be formulated by

\[ S_D = (Dv^T Dv)^{1/2} \]

where \( Dv = v(\eta_{k+1}) - v(\eta_k) \). From now on, the pseudo arclength of the vector of incremental change \( Dv \) is denoted with \( S \). This scalar can be used to define the step length such that the solution curve is properly presented and the computation is economical. The difficulty in determining the desired step length is that the part of the curve that has to be computed is not yet known. Consequently, it is only possible to use the history of the path calculated previously. From this history the curvature of \( C \) and the convergence behaviour of the corrector are important aspects for the determination of the step length. The curvature of the curve \( C \) can be determined by the second derivative of the function \( F(v, \eta) \). The convergence behaviour of the corrector is, for instance, characterized by the number of cycles that is needed to compute a satisfactory solution.

As far as accuracy is concerned, at each iterative cycle a trade-off must be made: firstly the iterative process should not be terminated too early because otherwise large deviations from the exact solution would occur, and secondly, the iterative process must not be allowed to go on indefinitely because that causes high computational costs. Consequently a second termination criterion is needed, complemented to the termination criterion to decide whether the iterates are diverging. This second termination criterion decides whether a solution is converged. Several options for the formulation of the criteria are possible. Here two formulations will be considered, the residual criterion and the error criterion. The residual criterion is the requirement that the norm of the residual force vector must be smaller than a prescribed value \( \| F(v^{k+1}) \| \leq \tau_f \). The error criterion makes use of the iterative improvement \( \| u^{k+1} - u^k \| \). Both of these can further be refined by using weighted versions of them. For details is referred to Ueberhuber [16].

### 3.1.1 Path-following process

The discussion of the principle of the path-following process will be followed by a discussion of the implementation of the method. The augmented system of equilibrium equations (3.2) is given by

\[ F(v, \eta) = \begin{bmatrix} f(v) \\ h(v) - \eta \end{bmatrix} = 0 \]

\[ h(v) : R_{N+1} \rightarrow R_1 \]

and a solution to this system (3.4) will simultaneously be an equilibrium state of the non-linear equations \( f(v) \), and a solution to the added equation \( f_{N+1} \). \( h(v) \) is the function for the auxiliary surface and, consequently, a solution to the system (3.4) is on this surface.
The solution curve $C$ is computed in a stepwise manner with the undeformed state $(u, \lambda) = (0,0)$ as the starting point. The sequence of successive steps, $v_k$, $k = 1, 2, 3, \ldots$ must be computed and for each step a new length $\Delta \eta$ is required. This length is, in general, dependent on the experience governed in the previous step. As follows from what has already been said, this suggests, e.g. to use the previous steplength $S_k$, the number of iterations of the previous step $i_{\text{prev}}$, and the curvature (embodied in $F_{vv}$) as indicators for the correct length of $\Delta \eta_{k+1}$. Thus

$$\Delta \eta_{k+1} = \Theta (S_k, F_{vv}, i_{\text{prev}})$$

The solution for step $k+1$, $v_{k+1}$, is the solution of the last step $v_k$ augmented by the changes following from the prediction and the correction process. For future reference, the total changes of an increment are denoted by $Dv$ while the iterative changes are denoted by $\Delta v$. Accordingly, $Dv = v_{k+1}^{i+1} - v_k$, and $\Delta v = v_{k+1}^{i+1} - v_{k+1}^i$. Herewith, the iterate $i+1$ for step $k+1$, $v_{k+1}^{i+1}$, is

$$v_{k+1}^{i+1} = v_k + Dv_{k+1}^{i+1}$$

$$Dv_{k+1}^{i+1} = \Delta v_{k+1}^{0} + \sum_{n=1}^{i+1} \Delta v_{k+1}^{n}$$

When the step length is determined, an approximation for the new solution $v_{k+1}$ can be made. This prediction is the previous solution augmented by a new step, denoted with $(\Delta v_{k+1}^{0}, \Delta \eta_{k+1})$. This step is in general derived from a relation

$$\Delta v_{k+1}^{0} = \Phi (v_k, (v_{\eta})_k, \ldots, v_{k-1}, (v_{\eta})_{k-1}, \ldots)$$

$$\Phi : R_{N+1} \rightarrow R_1$$

where $(v_{\eta})_k$ is the derivative of $v$ with respect to $\eta$, and $\Phi$ is a prediction function. Thus, the prediction can be viewed as a function of previous converged solutions, and their derivatives. The iterative process aims to improve the prediction and subsequent cycles, iterations, of the process, and is called the correction process. Accordingly, the corrector function $\Omega$ is defined in terms of the previous computed iterate, i.e. the iterate $v_{k+1}^i$. Accordingly, the change $\Delta v_{k+1}^{i+1}$ due to the correction can be written as

$$\Delta v_{k+1}^{i+1} = \Omega (v_{k+1}^{i})$$

$$\Omega : R_{N+1} \rightarrow R_1$$

This iterative process with the iteration function $\Omega$ is terminated when firstly, the subsequent iterates are diverging and secondly, the iterate is satisfying the error conditions. The two basic steps discussed in the foregoing results in a scheme of a prediction and a sequence of corrections. With this scheme one is able to compute points on the solution curve.

### 3.2 Solving the non-symmetric system

The discussion in this chapter is restricted to a linear prediction and correction process. Both, the prediction function, $\Phi$, and the iteration function, $\Omega$, are derived by applying Newton's method to the system of (3.4). More than that, in the discussion it will turn out that the corrector can even be used for the prediction. It makes sense therefore to consider the correction process first.
Newton's method is one of the most popular iteration schemes. When it is applied to the system of equation (3.4) to compute $v^{i+1}_{k+1}$ with the iterate $v^{i}_{k+1}$ as the starting point, the resulting system of equations yields

$$ F_v(v^{i}_{k+1}, \eta_k + \Delta\eta_{k+1}) \Delta v^{i+1}_{k+1} = -F(v^{i}_{k+1}, \eta_k + \Delta\eta_{k+1}) $$

where $F_v$ is the derivative of the function $F$ with respect to the vector $v$. To compute $\Delta v^{i+1}_{k+1}$, the inverse of this derivative is needed. From equation (3.5) it follows, that the iteration function $\Omega$ that belongs to this process contains the inverse of $F_v$. Notice, that the system $F$ is composed of the system $f$ and $f_{N+1}$ and in many engineering problems the stiffness matrix $f_v$ is symmetrical and sparse. A symmetrical and sparse matrix makes the calculation of its inverse in numerical analysis less expensive. But although the symmetry of the stiffness matrix is symmetric, the system as presented in equation (3.4) is in general non-symmetric. This lack of symmetry is due to the additional equation $f_{N+1}$. However, it is still possible to make use of the symmetry of the stiffness matrix by rewriting the system of (3.4). This is discussed in the following.

For reasons of simplicity in the notation, the subscript $k$ for the step index will be dropped as well as the superscript for the vector of changes $\Delta v^{i+1}$. Consequently, the correction is the difference between the approximated solution $v^i$, and the improved iterate $v^{i+1}$, $\Delta v = v^{i+1} - v^i$. Notice that with this convention the distinction between the notation of an arbitrary iterate $v^i$ and the solution $v$ still exists. The solution $v$ is the solution that is accepted as the sought solution. The residuals $F(v^i, \eta)$ and the Jacobian $F_v(v^i, \eta)$ are simply written as $F$ and $F_v$. In that case, equation (3.6) for the iterate $i+1$, is written as

$$ F_v \Delta v = -F $$

Newton’s method as given in equation (3.6) is actually a linearization of equation (3.4) around the approximation $v^i$. The derivative $F_v$, which is the Fréchet derivative of $F$, is the Jacobian matrix of the system and denoted by $J$. Herewith, equation (3.6) can be written in the form

$$ J(v^i) \Delta v + \begin{bmatrix} f(v^i) \\ h(v^i) - \eta \end{bmatrix} = J(v^i) \Delta v + \begin{bmatrix} r \\ f_0 \end{bmatrix} = 0 $$

where the vector $a = (r, f_0)^T$ represents the residuals of $F$ due to the fact that $f(v^i) \neq 0$ and the added equation $f_{N+1}(v^i) \neq 0$. These residuals are vanishing when the approximated solution $v^{i+1}$ reaches the solution for $\eta$ on the curve $C$.

For the present discussion $J$ is now written out in terms defined by

$$ J(v^i) = \begin{bmatrix} K & I \\ n^T & n_0 \end{bmatrix} $$

$$ K = f_u = \frac{\partial f}{\partial u} \in R_N \times R_N $$

$$ I = f_\lambda = \frac{\partial f}{\partial \lambda} \in R_N $$

$$ n^T = h_u = \frac{\partial h}{\partial u} \in R_N $$

$$ n_0 = h_\lambda = \frac{\partial h}{\partial \lambda} \in R_1 $$

40
K in here, is the stiffness matrix of the structural system, I the load vector and \((n^T, n_0)\) the normal to the auxiliary surface \(h(v)\). Although, the definition and the determination of this normal \(m^T = (n^T, n_0)\) turns out to be essential for the definition of the path parameter \(\eta\), it will be considered later. With the division of the Jacobian \(J\), it is possible to compute a solution out of equation (3.7), using the symmetry and sparseness of the stiffness matrix. Assuming that the system (3.4) is non-singular, so that the inverse of the Jacobian, \(J^{-1}\) and \(K^{-1}\), exists, the solution of equation (3.7) can be represented in two steps, as is formulated below.

\[
R = -K^{-1}r \\
L = -K^{-1}l \\
\Delta \lambda = \frac{n_0 + n^T R}{n_0 + n^T L} \\
\Delta v = \begin{bmatrix} \Delta u \\ \Delta \lambda \end{bmatrix} = \begin{bmatrix} R + \Delta \lambda L \\ \Delta \lambda \end{bmatrix}
\]

In this formulation \(R\) is the vector of displacements due to the residual forces \(r\), and \(L\) is the vector of displacements due to the total applied load \(l\) both for the iterate \(v\). This is a convenient and efficient way of calculating the corrector because it makes use of the sparseness and the symmetry of the stiffness matrix.

As mentioned, the scheme (3.9) is based on the assumption that \(J^{-1}\) and \(K^{-1}\) exists. These conditions are equivalent to the conditions:

\[
det [K(v)] \neq 0 \\
\tau = n_0 + n^T L \neq 0
\]

The second condition concerns the definition of the auxiliary surface and although it will be considered later, it can already be mentioned that this condition can always be satisfied as long as the auxiliary surface \(h(v)\) is properly chosen.

It should be mentioned that even when \(K\) is singular, and thus \(det[K] = 0\), the system (3.6) or (3.7) is still compatible when \(J^{-1}\) exists. However, the scheme (3.8) may then no longer appropriate and another way of solving system (3.7) must be sought. These points at which \(K\) is singular and \(J\) is nonsingular are points where the load \(\lambda\) reaches a stationary value. Consequently, these conditions define (among others) limit points. The singularity of \(K\) at these limit points does not imply, however, that the scheme (3.8) cannot be used in the corrector process of a path-following method that attempts to traverse a limit point. This is due to the circumstance that \(K\) is nonsingular, just before, and just after the limit point. Only in cases where the solution curve contains a stretch where the load \(\lambda\) satisfies \(\lambda = 0\), may lead to difficulties, Riks [14] Seydel [9].

A point at which \(K\) and \(J\) both are singular turns out to be a bifurcation point, Seydel [9] and Riks [6]. In this particular case, the path-following method that makes use of (3.8) will generally be able to step over this point in much the same way it steps over a limit point. However, for the computation of the branches of a bifurcation point additional measures are necessary. The discussion of these techniques fall beyond the scope of the present report.
3.3 Path-parameter

In the foregoing the path parameter $\eta$ was defined by the auxiliary surface $h(v)$ in the extra equation $f_{N+1}$. The added equation makes it possible to compute one isolated solution $v^*$ out of the set of solutions $\{v\}$ of the system (3.1). This set represents a one dimensional curve $C$ in the $N+1$ dimensional space. For a proper definition of $v^*$, it is necessary that the auxiliary surface $h(v)$ has an intersection with the curve $C$. According to the discussion of the incremental approach, the path parameter must be defined such that the following conditions are satisfied.

i. A step for the path parameter, $\Delta \eta_{k+1}$, corresponds to a solution that continues the previous computed part of the path. In other words, the step must be in the direction of interest.

ii. The augmented system of equations results in a system that is not ill-conditioned. This system gives for the step $\Delta \eta_{k+1}$ a change of the deformation which is comparable with the expected change. In other words a small step $\Delta \eta_{k+1}$ gives a small change of $u$, and vice versa.

iii. The solution $v_{k+1}$ for the step $(k+1)$ should reach the latter computed solution $v_k$ if the step $\Delta \eta_{k+1}$ goes to zero, thus when: $\Delta \eta_{k+1} \to 0$.

It will now be shown that these conditions are satisfied by the second relation of (3.7), thus by $m^T \Delta v - \Delta \eta = 0$.

The vector $m$, the normal to the auxiliary surface was introduced in (3.8). This object plays an important role in the conditions above and, consequently, in the following considerations. First, recall the added equation $f_{N+1} = h(v) - \eta = 0$, with the vector of unknowns which is parameterized by the path parameter $v(\eta)$. Let the auxiliary surface intersect with the curve $C$, for a range $\eta^- < \eta < \eta^+$. Differentiation of the extra equation, $f_{N+1}$, along the path yields

$$\frac{\partial}{\partial \eta} (f_{N+1}) = (m^T v') \left( \frac{dS_E}{d\eta} - 1 \right) = 0 \quad m = \left[ \frac{\partial h}{\partial v} \right] \quad v' = \left[ \frac{\partial v}{dS_E} \right]$$

3.11

where $v'$ is the unit tangent to the solution curve, or the so-called path derivative. It can now be observed that the product $m^T v'$ is given by $|m||v'| \cos \theta$. The term $dS_E/d\eta$ thus turns out to be inversely proportional with the cosine of the angle between the normal to the auxiliary surface $m$, and the tangent to the solution curve $v'$. This angle is denoted by $\theta$. To keep the change of $S_E$ with $\eta$ positive for increasing $\eta$, $\cos \theta$ must be positive and thus the angle between $m$ and $v'$ must be less than $\pi/2$ (figure 3-4). Consequently, the normal to the auxiliary surface $m$ must point in the direction of the path, for increasing $\eta$. The equivalent condition $m^T v' > 0$ guarantees the successive steps to be in the direction of interest.
Figure 3-5 Illustration of the linearization for a. the prediction and b. the correction.

Because \( \lim \| \eta \| = 1 \), equation (3.11) can be rewritten in the form

\[
dS_E = \frac{d\eta}{\cos \theta} \| v' \|
\]

and when \( \cos \theta \) reaches zero, a small change of the path parameter results in significant changes for \( dS_E \).

The limit case when \( \cos \theta \) is zero results in a rate of change of the arclength that becomes unbounded. This problem is equivalent with a previously mentioned problem, i.e. an ill-conditioned system where the auxiliary surface is parallel to the curve C. Thus, to prevent the system from becoming ill-conditioned, the angle between the normal \( m \) and the tangent \( v' \) should be acute. In this sense, the most ideal situation occurs when the normal \( m \) is parallel to the tangent \( v' \). The condition that \( m \) (at the point of intersection) should satisfy \( m^Tv' > 0 \), implying that the compatibility condition (ii) of section 3.2 is automatically satisfied because \( m^Tv' > 0 \) and \( n_0 + n^T1 \geq 0 \) are identical statements. This follows from the observation that \( (L,1)^T \) satisfies: \( f_{v'} (L,1)^T = KL + 1 \cdot 1 = 0 \), consequently, \( L \) corresponds with \( u' \) and thus \( v' = (u',1)^T \).

Next, the linearized system of equations, (3.7) with (3.8) is considered for the added equation \( f_{N+1} = h(v) - \eta \). Assume a general nonlinear auxiliary surface \( f_{N+1} \). The linearization of \( f_{N+1} \) at the solution \( v_k \) can be seen as the tangent plane in the solution \( v_k \) moved over a distance \( \Delta \eta \) along \( m_k \). This linearization of the added equation is given by \( f_{N+1} = m_k^T(v - v_k) + \Delta \eta_{k+1} = 0 \). The prediction \( v_{k+1}^0 = v_k + \Delta \eta_{k+1} m \) satisfies this equation. Equivalently, for the corrector, this linearization in the last computed approximation \( v^i \) is given by \( f_{N+1} = (m^i)^T(v - v^i) - \Delta \eta_{k+1} = 0 \). These linearizations are illustrated in figure 3-5, (a) for the case that this linearization is used for the prediction and (b) the linearization for the corrector. Notice that this 3D representation is only an illustration because in general the space is \( N+1 \)-dimensional and in figure 3-5a the residuals \( r \) in \( v_k \) are assumed to be zero.
As was mentioned before, there are infinitely many ways to define $f_{N+1}$ (see also Riks 1970 [10], Riks 1972 [11]). It is possible to make a distinction between (i) surfaces $f_{N+1}$ that are varied independently of the solution curve $C$, (ii) surfaces that are changed in each or some steps along $C$ taking into account the development of the path and (iii) surfaces that are changed in each step and each cycle of the corrector process. Another distinction that can be made is that one can choose between (I) linear and (II) nonlinear functions for $f_{N+1}$ in $v$.

Examples of choices of the first category (i) were already given; the load stepping procedure and the displacement stepping procedure, discussed in section 3.1. In these cases one fixed degree of freedom is prescribed for each step. For reasons already explained these choices are not very appropriate for general applications. The question thus remains which of the other options should have a preference. The difference between the two other options (ii) and (iii) is the definition of $f_{N+1}$ in the correction process. For the second category (ii) the surface $f_{N+1}$ of the beginning of the step is used and for the third category (iii) the surface is updated for each cycle of the process. Therefore the third category may only be of an advantage when the direction of the solution curve $C$ changes significantly in one step.

It can now be observed that the path following principle is based on the assumption that the steps that are taken along the solution curve are small for reasons discussed before. Consequently, the change of the direction of the curve $C$ between two steps is almost always very small i.e., $||\Delta \eta||^2 ||v'||^2 << ||\Delta \eta||^2 ||v'||$. Therefore, the effect of taking into account the curvature in one step, is insignificant. This means that the difference between the remaining classes of functions for $f_{N+1}$, (ii) and (iii), will have no impact on the overall outcome and efficiency of the computations. This conclusion also holds for the supposed advantage of a nonlinear function (II) $f_{N+1}$ against a linear form (I). The difference of the latter can be seen from figure 3-5, where the tangent plane is the linearization of the general nonlinear surface $f_{N+1}$.

The selection of $f_{N+1}$ should therefore be based on the simplicity of the formulation, a conclusion that was followed in Riks [10] and Riks [13] to formulate the following auxiliary surface of category (ii,I):

$$ f_{N+1} = m_k^T (v - v_k) - \Delta \eta = 0 $$

where $m_k$ is taken to be the tangent to the path at the previous step $k$, $m_k = v_k$, or a suitable approximation to $v_k$. In this choice, see figure 3-5, the length $\Delta \eta$ can be seen as an approximation of the actual arclength, $\Delta s$, measured between the point $v_k$ and the still undetermined new point $v_{k+1}$. Hence the term pseudo arclength that is associated with this choice.

Another popular pseudo arclength is associated with a choice that belongs to category (ii,II). This is the surface:

$$ f_{N+1} = (v - v_k)^T (v - v_k) - (\Delta \eta)^2 = 0 $$

which is identified as a sphere in $R_{N+1}$ with radius $\Delta \eta$. This choice, suggested by Crisfield [3], is introduced with the aim to improve the choice introduced by Riks (3.12) but it can be mentioned that this (nonlinear) alternative has no impact on the overall outcome and efficiency of the computations. It often cause some difficulties, the nature which will be discussed at an other place.
3.4 Implementation of the path-following method in MARC

In the MARC code, two choices for the path-following method are implemented: The first is the implementation as suggested by Riks, and the second, the implementation as suggested by Crisfield. The difference between these methods is in the definition of the added equation $f_{N+1}$. This equation describes the auxiliary surface and the versions Riks and Crisfield proposed, were already mentioned in section 3.3, in equation (3.12) and (3.13) respectively. Both implementations of the path-following method use a simplified equation $f_{N+1}$. This simplification consists of neglecting the contribution of the load term $\lambda$ in $\nu = (u, \lambda)$. It should be noted that equation (3.13) is a spherical constraint. Consequently, by neglecting the load term, the added equation for Crisfield becomes a cylindrical constraint.

The implementation in MARC makes use of the full Newton-Rhapson scheme for the iterative process. In case of full Newton-Rhapson, the stiffness matrix is factorized in each cycle of the iterative process.

The first method that will be considered now is the implementation according to Riks [10] (In the literature the particular version used here is often ascribed to Ramm [9] but this is incorrect because the variation introduced by Ramm is minor and above all without any particular effect).

3.4.1 Riks' implementation of path-following

The implementation of the path following method according to Riks is based on the derivation given at (3.6) - (3.9). The system $f(\nu) = 0$ in equation (3.1) is augmented with $f_{N+1}$ to have a determined system of equations, equation (3.2). To be able to compute a loadstep $\Delta \lambda$ and corresponding displacements $\Delta u$, the augmented system $F(\nu; \eta) = 0$ is linearized at the last derived solution $\nu_k$, equation (3.7). Equation (3.9) gives the resulting correction for $\Delta \lambda$ and $\Delta u$. The corrector step for iteration count $i+1$ of step $k+1$ can be written as,

$$
\Delta \lambda = \frac{r_0(u^i) + n^T R(u^i)}{n_0(u^i) + n^T L(u^i)}
$$

$$
\Delta \nu = \begin{bmatrix} \Delta u \\ \Delta \lambda \end{bmatrix} = \begin{bmatrix} R + \Delta \lambda L \\ \Delta \lambda \end{bmatrix}
$$

(3.14)

However, this relation can also be used to construct a prediction along the tangent vector of curve $C$ in $\nu_k$ (figure 3.6). Recall that the tangent $\nu'_k$ is described by the system

$$
F'_\nu(\nu_k) \nu'_k = \begin{bmatrix} f'_\nu(\nu_k) \nu'_k \\ h'_\nu(\nu_k) \nu'_k - 1 \end{bmatrix} = \begin{bmatrix} K(\nu_k) u'_{k+1} + \lambda'_{k+1} \\ n^T u'_{k+1} + n_0 \lambda'_{k+1} - 1 \end{bmatrix} = 0
$$

(3.15)

using

$$
\Delta u^0_{k+1} = \Delta \eta_{k+1} u'_{k+1}
$$

$$
\Delta \lambda^0_{k+1} = \Delta \eta_{k+1} \lambda'_{k+1}
$$

in (3.15) gives
Continuation Methods for Contact Analysis

Figure 3.6 Prediction of step $k+1$ along the tangent

$$K(v_k) \Delta u_{k+1}^0 + l \Delta \lambda_{k+1}^0 = 0$$

$$n^T \Delta u_{k+1}^0 + n_0 \Delta \lambda_{k+1}^0 = \Delta \eta_{k+1}$$

The latter system can be solved by (3.14) the following condition

$$r_0 = -\Delta \eta_{k+1}, \ n_0 \equiv 0, \ n = n(v_k)$$

$$R = R(v_k) = 0, \ L = L(v_k)$$

The normal to the auxiliary surface $m$ in equation (3.12) reduces to $n$ when the load term $\lambda$ is neglected. Equation (3.12) becomes thus $f_{N+1} = n_k^T \{u - u_k\} - \Delta \eta_{k+1} = 0$. In the MARC-code the normalized displacements of the previous step are chosen for the normal to the auxiliary surface and thus

$$n^T = h_u = \frac{[Du_k]^T}{\|Du_k\|} \Rightarrow Du_k = u_k - u_{k-1}$$

Furthermore, because the load term is neglected in the extra equation, $n_0 = 0$, $\Delta \eta_{k+1}$ is the pseudo arclength defined by the norm of the displacements $\|Du_{k+1}\| = (Du_{k+1}^T Du_{k+1})^{1/2}$. In the conditions (3.17) the residuals are assumed to be "small", $R \ll \Delta \eta = 0$. This is because the iterative process is not terminated until the residual error is sufficiently small. Unless $R$ is almost zero at the end of the previous increment, it is not neglected in the relation for the prediction in the MARC-code. This means that the prediction is calculated by

$$\Delta \lambda_{k+1}^0 = \frac{\|Du_k\| \Delta \eta_{k+1} - [Du_k]^T R^0}{[Du_k]^T L^0}$$

$$\Delta \lambda_{k+1}^0 = \begin{bmatrix} \Delta u_{k+1}^0 \\ \Delta \lambda_{k+1}^0 \end{bmatrix} = \begin{bmatrix} R^0 + \Delta \lambda_{k+1}^0 L^0 \\ \Delta \lambda_{k+1}^0 \end{bmatrix}$$
For the corrector, equation (3.9) can directly be applied when

\[ r_0 = -\Delta n_{k+1}, \quad n_0 = 0, \quad n = n(v_{k+1}^i) \]

\[ R = R(v_{k+1}^i), \quad L = L(v_{k+1}^i) \]

is used. Riks pointed out that the normal to the auxiliary surface \( n \) need only be changed once at each continuation step for simplicity. This choice for the auxiliary surface \( f_{N+1} \) belongs to category (ii, I) discussed in section 3.3. According to this choice, the displacements of the previous step \( Du_k \) are used in each cycle of the correction process. However, for ease of implementation in the MARC-code \( n \) is updated in each iteration cycle by setting

\[ n^T = h_u = \frac{[Du_{k+1}^i]^T}{[Du_{k+1}^i]} \quad \Rightarrow \quad Du_{k+1}^i = \{u_{k+1}^i - u_k\} \]  

3.19

The latter is thus the variation Ramm proposed. It is seen that \( f_{N+1} \) is updated in each cycle so that \( f_{N+1} \) belongs to category (iii, I, section 3.3). The assumption will be made that the residual to the auxiliary surface \( h(v) - \eta = 0 \) is small \( v_0 \ll \Delta \eta \) and can be neglected. The relations to compute \( \Delta \lambda_{k+1}^{i+1} \) and \( \Delta v_{k+1}^{i+1} \) are then given by

\[ \Delta \lambda_{k+1}^{i+1} = \frac{[Du_{k+1}^i]^T R^i}{[Du_{k+1}^i]^T L^i} \]  

3.20

\[ \begin{align*}
\Delta v_{k+1}^{i+1} = & \begin{bmatrix}
\Delta u_{k+1}^{i+1} \\
\Delta v_{k+1}^{i+1} \\
\Delta \lambda_{k+1}^{i+1}
\end{bmatrix} =
\begin{bmatrix}
R^i + \Delta \lambda_{k+1}^{i+1} L^i \\
\Delta \lambda_{k+1}^{i+1}
\end{bmatrix}
\end{align*} \]

Consequently, in MARC, two small variations of Riks’ method are implemented: (i) the displacements due to the residual forces \( R \) are not neglected in the prediction and (ii), the normal to the auxiliary surface is adapted in each cycle of the iterative process.

In general, the displacement component due to the residual \( R \) is “small” at the beginning of an increment, \( \|R\| \ll \Delta \eta \). But when the second term in the nominator of (3.18) becomes in the order of the first term, it follows that the step in the prediction gets smaller, and even becomes approximately zero. When the second term of the nominator of (3.18) is neglected and the prediction is along the tangent in \( v_k \), this difficulty never occurs. Therefore, \( R \) should be set to zero.

As mentioned before, the auxiliary surface, \( f_{N+1} \), can be seen as a control surface for the iterative process. When the normal \( n \) is adapted in each cycle of the correction process, the control surface is changed as well. This influences the convergence behaviour, and therefore, \( Du_k \), must be used for the normal for each cycle of the iterative process. According to this considerations, (ii) is not an improvement and it turns out that this particular problem occurs when the norm of the displacement vector due to the residual forces \( \|R\| \) is of the order \( \Delta \eta \).

The implementation in the MARC code is suitable when the norm of the displacements due to the residuals, \( \|R\| \), is considerably smaller than the step length \( \Delta \eta \). It can be improved by the mentioned measures or by setting the convergence criterion tighter. The procedure is summarised in appendix A-2.
3.4.2 Crisfield's implementation of path-following

The implementation of the path-following method according to Crisfield [3] is based on the linearization of the equilibrium conditions, \( f(v) = 0 \) in equation (3.1), augmented with a nonlinear equation \( f_{N+1} \). This nonlinear equation (3.13) for Crisfields implementation belongs to category (ii,II) and is discussed in section 3.3. The resulting set of corrector equations is

\[
f(v_k) + f_r(v_k) \Delta v = 0
\]

\[
f_{N+1} = (v - v_k)^T (v - v_k) - (\Delta \eta)^2 = 0
\]

3.21

For the predictor one can use

\[
K(v_k) \Delta u_{k+1} + I \Delta \lambda_{k+1} + r(v_k) = 0
\]

\[
[D u_{k+1}]^T D u_{k+1} - (\Delta \eta_{k+1})^2 = 0 \quad \Rightarrow \quad D u_{k+1} = u_{k+1} - u_k
\]

3.22

where the load term in \( f_{N+1} \) is neglected and the notation of equation (3.7) and (3.8) is followed. Equation (3.22) describes a step, with length \( \Delta \eta_{k+1} \), along the tangent. Substitution of \( \Delta u_{k+1} \) from the first equation of (3.22) in the latter gives a quadratic equation in \( \Delta \lambda_{k+1}^0 \).

\[
a_1 (\Delta \lambda_{k+1}^0)^2 + a_2 \Delta \lambda_{k+1}^0 + a_3 = 0
\]

\[
\Delta \gamma_{k+1}^0 = \begin{bmatrix} \Delta u_{k+1}^0 \\ \Delta \lambda_{k+1}^0 \end{bmatrix} = \begin{bmatrix} R + \Delta \lambda_{k+1}^0 \lambda \\ \Delta \lambda_{k+1}^0 \end{bmatrix}
\]

3.23

where the coefficients \( a_1, a_2 \) and \( a_3 \) are defined by

\[
a_1 = [L]^T L, \quad a_2 = 2 [L]^T R, \quad a_3 = [R]^T R - (\Delta \eta)^2
\]

This quadratic equation has two roots. Consequently, an extra step is needed to compute the loadstep \( \Delta \lambda \) and the corresponding displacements \( \Delta u \). These two roots correspond to the two intersections of the tangent with the cylindrical constraint, \( f_{N+1} \). However, it is questionable whether the use of the quadratic equation in (3.22) and (3.21) is really essential. In relation to this, the following two remarks are made.

- It is not necessary, during the iterations, to satisfy the auxiliary surface exactly.
- The quadratic equation introduces two solutions. Furthermore it is not certain at all that real roots exist (in particular in the case of the corrector equation (3.21)).

Equation (3.23) results in two roots for \( \Delta \lambda \) which corresponds to displacements along the tangent, in the intended direction and in the opposite direction. According to the discussion of the normal to the auxiliary surface, \( n \), one can test on the inner product of \( D u_{k+1} \) and a normal \( n \). The angle between the change of the displacements of the previous increment, \( D u_{k+1} \), and the tangent \( u \) is small when the steps are not too large relative to the curvature of the solution curve C. This means that one can test on the inner product to be

\[
[D u_{k}]^T D u_{k+1} > 0
\]

3.24
The iterations of the corrector process can be calculated by (3.22) when the new iterate $v^{i+1}$ is substituted for the new step $v_{k+1}$ and the previous iterate $v^i$ for the previous increment $v_k$. Thus (3.22) becomes for the corrector

\[
K(v_i) \Delta u^{i+1} + I(\Delta \lambda^{i+1}) + r(v^i) = 0
\]
\[
[D u^{i+1}]^T D u^{i+1} - (\Delta \eta^{i+1})^2 = 0
\]
\[
\Rightarrow D u^{i+1} = u^{i+1} - u_k = D u^i + \Delta u^{i+1}
\]

Substitution of $\Delta u^{i+1}$ from the first equation of (3.22) in the latter gives a quadratic equation in $\Delta \lambda^{i+1}$.

\[
a_1 (\Delta \lambda^{i+1})^2 + a_2 \Delta \lambda^{i+1} + a_3 = 0
\]
\[
\Delta v^{i+1} = \begin{bmatrix} \Delta u^{i+1} \\ \Delta \lambda^{i+1} \end{bmatrix} = \begin{bmatrix} R^i + \Delta \lambda^{i+1} L_i \\ \Delta \lambda^{i+1} \end{bmatrix}
\]

where the coefficients $a_1, a_2$ and $a_3$ are defined by

\[
a_1 = [L^i]^T L^i
\]
\[
a_2 = 2 [L^i]^T Du^i + 2 [L^i]^T R^i
\]
\[
a_3 = [D u^i]^T Du^i + 2 [D u^i_{k+1}]^T R^i + [R^i]^T R^i - (\Delta \eta)^2
\]

The root in the intended direction can be determined by

\[
[D u^i]^T D u^{i+1} > 0
\]

When both roots corresponds to positive products Crisfield suggested to choose the loadstep $\Delta \lambda$ corresponding to the displacements closest to the linearised solution of equation (3.23), i.e.:

\[
\Delta \lambda^{i+1} = \frac{a_3}{a_2}
\]

When equation (3.23) or (3.26) has no real roots, Crisfield suggests to reduce the steplength. On the other hand it will turn out later that in combination with mechanical contact a reduction of the steplength, as recommended by Crisfield, does not solve the problem of imaginary roots. In the appendix A-1 a geometrical interpretation of this situation is discussed. This is done, because in the literature, the situation where no real roots of the quadratical equation exists is often explained incorrectly.
3.5 Conclusions

This chapter contained a discussion of the path-following method. The assumptions on which this method is based are: (i) that the set of solutions that satisfies the system (3.1), the curve $C$, is continuously differentiable, (ii) that no bifurcation points in the curve $C$ occur and (iii) that the number of unknowns is not changed during the analysis.

Bifurcation points (ii) do not occur when condition (i) is fulfilled but in practice the path-following procedure can jump over the bifurcation points. When this happens it is necessary to analyse the bifurcation point. For such an analysis the reader is referred to e.g. Riks [10]. The condition (iii) that the number of unknowns of the system remains constant during the analysis seems obvious but in the next chapter it will be shown that when contact occurs, the reduction of the number of unknowns due to contact must be considered.

The path-following method is implemented according to Riks and according to Crisfield. It is assumed that in general the norm of the displacements due to the residuals, $\|\mathbf{r}\| = \| - \mathbf{K}^{-1} \mathbf{r} \|$, is considerable smaller than the step length $\Delta \mathbf{n}$. When this norm $\|\mathbf{r}\|$ becomes in the order of $\Delta \mathbf{n}$ it is not certain any more that the procedure converge to a solution and it is therefore necessary to assure that the termination criteria are in accordance with this requirement.

To facilitate the implementation of Riks’ method in the MARC code, the displacements of the last iterate are used for the normal to the auxiliary surface, instead of the displacements of the previous step. However, it is preferable to use the displacements of the previous step $\mathbf{du}_k$ for each cycle in the iterative process. For the implementation of the prediction of Riks’ method in MARC, the displacements due to the residuals are not neglected. Both of these choices can cause problems in particular when the norm $\|\mathbf{r}\|$ becomes large, relative to $\Delta \mathbf{n}$.

The nonlinear augmentation in Crisfield’s method, $f_{N+1}$, gives two roots. Therefore it is necessary to decide which root is in the intended direction. The problem of a quadratic equation can become worse when the norm of the displacements due to the residuals, $\|\mathbf{r}\|$, is large, relative to the step length, $\Delta \mathbf{n}$. In that case, the process may fail because no real roots exists, and therefore, this implementation is actually not suitable for applications which are strongly nonlinear such as problems that describe contact.
Chapter 4

Continuation methods for contact analysis

4.1 The tying relations in path-following methods

In chapter 2, the tying method is discussed. It is characteristic for this method that the displacements of the hitting nodes are constrained to the displacements of the target nodes (equation (2.28)) and, consequently, when contact takes place, the number of unknowns is reduced. Thus, the system of equilibrium equations before contact occurs (2.6) and (3.2), $F : R_{N+1} \rightarrow R_{N+1}$ (referred to as the full system of equations) changes in that case to $F : R_{N+1-n} \rightarrow R_{N+1}$ after contact occurs, where $n$ is the number of tyings that is created. The system after contact is so called overdetermined. This implies that the number of equilibrium equations must be reduced such that $\tilde{F} : R_{N+1-n} \rightarrow R_{N+1-n}$, where $\tilde{F}$ is the reduced system of equations (2.24). The reduction can be obtained by pre-multiplying the system of equation (2.6) by the tying matrix $S$. Consequently, the system changes for each new hitting node. The forces on the hitting nodes are computed from the full system of equations (2.2). Consequently, when these forces are needed, the system must be transformed to the full system in each increment. The moment at which this back transformation should be carried out is a point of consideration. It can be done before or after computing the incremental or iterative changes $\Delta \lambda^i$ and $\Delta u^i$. Because it is preferable to use a system of equations of the same dimensions during the analysis, it is expedient to transform the reduced system to the full system before the calculation of the new step or iteration is carried out.

This moment of transformation, in advance of the computation of the iterative change, will now be verified for the Riks implementation of path-following. Therefore, in contrast to the discussion above, the path-following relations will be transformed to the reduced system, $R_{N+1-n}$. The vector $\Delta u = S \Delta \tilde{u}$ from equation (2.23) is substituted in $-K\Delta u = r + \Delta \lambda l$, (3.7), resulting in a system which is overdetermined. Reduction of this system gives a relation in terms of the reduced displacement vector $\Delta \tilde{u}$

$$-\tilde{K}\Delta \tilde{u} = \tilde{r} + \Delta \lambda l \quad \Rightarrow \quad \Delta \tilde{u} = \tilde{R} + \Delta \lambda \tilde{L}$$

where:

$$\tilde{r} = S^T r, \quad \tilde{R} = -\tilde{K}^{-1} \tilde{r}, \quad \tilde{L} = -\tilde{K}^{-1} l$$

where, $\tilde{r} = S^T r$, $\tilde{R} = -\tilde{K}^{-1} \tilde{r}$ and $\tilde{L} = -\tilde{K}^{-1} l$. $\tilde{K}$ is the reduced stiffness matrix in these relations. Notice that the reduced vectors $\tilde{R}$ and $\tilde{L}$ are found by substituting the reduced matrices and vectors in (3.9). The reduced displacement vector is transformed to the full vector by $\Delta u = S \Delta \tilde{u}$ and substitution in $n^T \Delta u + r_0 = 0$ yields
$$\Delta \lambda = \frac{r_0 - n^T S \tilde{R}}{n^T S L} = \frac{r_0 - n^T R}{n^T L} \quad 4.2$$

It follows that the displacements due to the residuals and the load $\tilde{R}$ and $\tilde{L}$ must be pre-multiplied by the tying matrix $S$ before these vectors can be used in the computation of the load step $\Delta \lambda$. This implies that transformation of the reduced system to the full system in advance of the computations for path-following process is preferable. Substitution of the $\Delta \lambda$ in the relation for the change of the displacements $\Delta u = R + \Delta \lambda L$ gives the unknown displacements.

The products in the denominator and the last term of the nominator of equation (4.2) cannot be carried out when the dimensions of the vectors $n^T$, $R$ and $L$ are not equal. Therefore, it is necessary to convert all the vectors to the full or to the reduced system. Because, in general, the vector $n^T$ is computed in the previous step or iteration, it is preferable to take the full vectors as reference because this leads to a reduction in computation. Furthermore, the calculation of the incremental changes for the full system has the advantage that $\Delta \eta$ does not need to be transformed. To show the difficulties for this transformation and thus the difficulties that arises for the definition of the arclength when the reduced solution vector is used will now be considered.

The increment $\Delta \eta$ corresponds to the incremental change of the displacement vector $\Delta \eta = ||Du||$. However, in the reduced system we have $||D\eta|| \neq ||Du||$. Calculation of the incremental changes with the reduced vectors and thus with a reduced arclength leads to the difficulty that the arclength must be converted. The arclength of the previous increment $\Delta \eta_k$ can be computed with $\Delta \eta_k = ||S_{k+1}Du_k||$, if the full vector $Du$ is known. But when the reduced system is used this relation cannot be used. Consequently, additional vector products are required which is undesirable. On the other hand when the full system is used for the computation of the incremental changes, these difficulties do not arise.

Equation (4.2) is directly applicable for the prediction in the Riks method. By neglecting the residual $r_0$, the equivalent relation for the correction is derived. With a similar approach, the $\Delta \lambda$ for Crisfield’s implementation can be computed by converting the reduced vectors to the full vectors, and subsequently substitute them in equation (3.23).

The constraints for the configuration in rigid contact analysis are the fixed displacements imposed by the displacements of the rigid body as mentioned in chapter 2. When the nodes are in contact, the displacements of the hitting nodes normal to the rigid body surface are described by the displacements of the rigid body. Notice that the rank of the system remains unchanged. To satisfy the constraints, the system is modified by adding large values to the corresponding terms of the stiffness matrix and the force vector so that these additions prescribe the displacements of the hitting nodes. When the incremental motion of the rigid body is known, the displacements of the nodes $\Delta u_c$ in contact can be computed and the corresponding step $(\Delta u, \Delta \lambda)$ follows from equation (2.25). When $(\Delta u, \Delta \lambda)$ does not exactly fulfill the condition of equation (2.25), a residual force $r$ is introduced, which is equivalent to the linearisation in equation (3.7) and (3.8). This results in

$$\begin{bmatrix} K + \sum_{n=1}^{m} \alpha e_n e_n^T \end{bmatrix} \Delta u + \Delta \lambda \lambda + \alpha \Delta u_c + r = 0 \quad 4.3$$
where the terms in the sum on the left hand side, $\alpha e_n e_n^T$, are the values which are added to the stiffness matrix, and the second last term $\alpha \Delta u_c$ is the corresponding fixed displacements due to contact. When $\alpha$ is large relative to the maximum value of the terms in the stiffness matrix, thus $\alpha \gg \max(k_{ij})$, the constraints will be satisfied. A difficulty in equation (4.3) is that when the path-following method is used, $\Delta u$ and $\Delta \lambda$ are unknown in advance. Consequently, the displacements $\Delta u_c$ due to contact with the rigid body cannot be determined. However, when the displacements of the rigid body are transformed to an externally applied fixed displacement, these displacements can be added to the loadvector. Therefore, it is necessary to compute the displacements of the hitting nodes in contact which are due to the total displacements of the rigid body. These displacements $\Delta u_{clot}$ are then scaled by $\Delta \lambda$ when (4.3) is changed to

$$
\left[ K + \sum_{n=1}^{m} \alpha e_n e_n^T \right] \Delta u + \Delta \lambda (1 + \alpha \Delta u_{clot}) + r = 0
$$

4.4

4.1.1 Nonhomogeneous tying relations

In chapter 2, tolerances on the target surface are introduced to control the errors which are introduced by the discretisation and the linearisation of the nonlinear process. Furthermore, tolerances reduce the number of splittings necessary to make a step $\Delta \eta$. This reduction is caused by the number of nodes that is within the tolerance but not penetrating the target body. These hitting nodes are tied to the target body at the start of the increment and thus the splitting process for these nodes is not needed. When the hitting nodes within the tolerance must be tied to the target body, they first need to be moved to the target surface (figure 4-1). These displacements appear as a nonhomogeneous term in equation (2.30). The difficulty of moving the hitting nodes to the target segments, once the distances between them has reached a certain tolerance, will be discussed by considering the implementation of the Riks method. The resulting technique can also be applied to Crisfield's method.

According to equation (2.29) the relation for the incremental displacements in the local directions is given by

$$
\Delta u_{kl}(\xi, \eta) = \phi^{\omega}(\xi, \eta) \Delta u_{\text{el}} + \Delta u_{\text{NH}}
$$

4.5
where $\Delta u_{NH}$ is the displacement vector that moves the hitting nodes to the target segment. Equation (2.30) is rewritten by using (2.23) giving

$$\Delta u^* = S\vec{u} + \Delta u_{NH}$$  \hspace{1cm} (4.6)

where the vector of the total incremental displacements $\Delta u^*$ is composed of the vector for the incremental displacements $\Delta u = S\vec{u}$ and the displacements due to the (forced) tying $\Delta u_{NH}$. Substitution in equation (4.1) gives

$$-S^T K (S\vec{u} + \Delta u_{NH}) = S^T r + \Delta \lambda S^T 1 \Rightarrow -\bar{K}\bar{u} = r^* + \Delta \lambda 1$$  \hspace{1cm} (4.7)

where $\bar{K} = S^T K S$, $I = S^T 1$ and $r^* = S^T (r + K\Delta u_{NH})$. The vector $r^*$ is the vector of the residuals at the start of the increment plus the vector of forces that are due to the non-homogeneous displacements $K\Delta u_{NH}$. These forces change the residuals and, obviously, are the forces that are required to "pull" the hitting nodes to the target surface. This vector will never be scaled and the iterative process aims to reduce these forces. When changes in the contact conditions are carried out in the prediction, the basic correction formula can be used. Substitution of equation (4.7) in equation (4.6) gives the total incremental displacement vector.

In rigid contact analysis the imposition of contact is performed by describing the displacements that follow from the rigid body motion. The non-homogeneous terms can be taken into account in the same way as is done in equation (4.7) but without the transformations due to the tyings. In rigid contact the constraint displacements will change the stiffness matrix. Therefore, the incremental displacements for rigid contact yields

$$\Delta u^* = \Delta u + \Delta u_{NH}$$

$$-K\Delta u = (r + K\Delta u_{NH}) + \Delta \lambda 1$$

where the vector of the total incremental displacements $\Delta u^*$ is composed of two terms, the term for the incremental displacements $\Delta u$ due to a load step $\Delta \lambda$ added by the displacements $\Delta u_{NH}$ due to the tolerance to the rigid target surface $\Delta u_{NH}$. Applying this to equation (4.4) gives

$$- \left[ K + \sum_{n=1}^{m} \alpha e_n e_n^T \right] \Delta u = (r + K\Delta u_{NH}) + \Delta \lambda (1 + \alpha \Delta u_c)$$

This completes the discussion about the implementation of the modelling of contact in the path-following method. Next, some difficulties that occur in the use of the method in contact analysis are considered.
4.2 Discontinuity of the path

In chapter 3, the solution to system (3.2) is considered to be a one dimensional solution curve C. This curve is also referred to as the path of system (3.2). Discontinuity in the path derivatives of C, and therefore the stiffness of the configuration, is the main characteristic of contact analysis. This means that in contact analysis the equilibrium path has singular points $v^s$. At the singular points $v^s$ the path is still continuous contrary to its derivatives. In these points, the inverse of the stiffness matrix $K^{-1}(v^s) = [f_0(v^s)]^{-1}$ does not exist. Therefore, the path is piece wise continuously differentiable and the analysis necessitates a restart in every singular point $v^s$.

Discontinuity of the stiffness occurs for two reasons; (i) the geometrical discontinuity and (ii) the discontinuity due to the discretisation of the contacting bodies. The geometrical discontinuity indicates the difficulty of a sudden change of the geometry due to contact. Geometrical discontinuity occurs e.g. when a soft body is contacting a relatively stiff body. The freedom to move for the soft body will be much less when contact occurs. The second reason for discontinuity is the discretisation of the contacting bodies, and thus, a discretised contact surface. This discontinuity has less impact on the curve C but the stiffness of the structure is discontinuous each time a node is contacting. Therefore the path is contains a kink each time a hitting node contacts the target surface, or when a hitting node slides from one target segment to another.

The path in the neighbourhood of the singular point $v^s$ can be divided in a set of solutions before the singularity denoted by $\langle v, v^s \rangle$ and a set of solutions after the singularity denoted by $| v^s, v \rangle$. $\langle v, v^s \rangle$ stands for the set of solutions v before contact occurs and bounded by the singular value $v^s$. When the neighbourhood is assumed to be small, e.g. $\|v^s - v\| \ll Dv$, the stiffness matrix $K$ is nearly constant in the set of solutions $\langle v, v^s \rangle$ and $| v^s, v \rangle$. Because of the singularity, however, the stiffness matrix before and after the singular point $v^s$ is different $K_{\langle v, v^s \rangle} \neq K_{| v^s, v \rangle}$. Notice that for the tying method the process must be stopped to allow the construction of the tying line.

The path-following method as derived in the previous chapter is based on the assumption that the path is continuously differentiable, and thus smooth. Therefore the method cannot directly be used at the singular points $v^s$. In between these points, the use of the method is standard. Required for the continuation of the path in a singular point $v^s$ are (i) the starting value, (ii) the new tangent for $| v^s, v \rangle$, and (iii) the continuation of the process must be in the intended direction. The latter is fulfilled automatically with our choice of the predictor. Because the path itself is continuous, the values for the displacements and the external forces should be equal before and after $v^s$ and, accordingly, condition (i) for the continuation is fulfilled by using the singular point $v^s$ as the starting point. In section 3.3 it is mentioned that the tangent to the path can be written as $v = [u', 1]^T = [L, 1]^T$. When the load term in the latter relation is neglected, L can be seen as the tangent to the path. Because $L$ is defined by $L = -K^{-1}1$, the tangent does not exist uniquely at the singular points. In this singular point a node hits the target body and, consequently, the hitting node should be tied. Because of the tying, the stiffness matrix $K$ changes and with the changed stiffness matrix $K_{| v^s, v \rangle}$ the tangent for the path after the singularity changes from $L_{\langle v, v^s \rangle}$ to $L_{| v^s, v \rangle}$ and thus exists (condition (ii) for the continuation).
4.2.1 Increment splitting

When a node hits the target body at an arbitrary point along a new increment, the analysis must be restarted at this point of contact. Accordingly, the increment can be reduced or split into parts for each contacting node. Reduction of the increment is followed by the calculation of a new step $\Delta \eta$. When $\Delta \eta$ is such that in that step several nodes are hitting the target body, this reduction leads to increments which are relative small to the arclength $\Delta \eta$. These small increments may lead to costly computations and therefore it is preferable to split the increment in a number of subincrements and try to reach the new solution within the defined step length $\Delta \eta_{k+1}$. With relatively large increments or fine meshes, this results in a number of subincrements for the step $\Delta \eta$. In this way, splitting the increment corresponds to the ordinary incremental approach with a constant or gradually changing arclength $\Delta \eta$. After splitting, either the hitting node can be tied for deformable contact or the displacements of the hitting node can be prescribed by the motion of the rigid body, for rigid contact.

Before an increment can be split, a fictitious solution and a scaling factor are required. This fictitious solution is a solution at which penetrating nodes are allowed. The scaling factor $\alpha$ is determined by the penetration of the node with the maximum penetration for the fictitious solution. After that, the increment is scaled in a way that this hitting node is just on the surface of the target body. In linear analysis it is possible to scale the increment such that the penetrating node is exactly on the target surface. However, in nonlinear analysis this is not possible. When the hitting node is not exactly on the target surface $g \neq 0$, it is forced to the target surface so that residual forces $r$ are introduced. These residual forces can be seen as the forces that keep the hitting node at the target surface. These nonhomogeneous type of tyings are discussed in section 4.1. In nonlinear analysis, the magnitude of the residuals depends on the accuracy of the scaling procedure and when this accuracy is low, the residuals may become large. In that case, the number of iterates required to reach a solution increases. In the following discussion it will be assumed that linear scaling only introduces small residuals which eventually disappear in the iterative process.

Next, the procedure of increment splitting is considered. Suppose that at a certain step $k+1$ penetration is detected, i.e. nodes have penetrated the target body. Suppose further that this fictitious state has been computed successfully by means of the increment $\Delta d_i = \Sigma \Delta d_{i+1}$ (summation over $i$). To determine the point of contact within this increment, increment splitting must be carried out. This can be done by setting $d_i = \alpha (\Delta d_i)$ in which $d_i$ is the displacement vector of the first subincrement in increment $k+1$. Sometimes, one time splitting is not sufficient. If there are more than one necessary, the notation $d_j$ is used where $j$ is the index that determines the number of the subincrement. The rule for the splitting is in that case $d_{j+1} = d_j + \alpha_{j+1}(\Delta d_j)$. After the increment is split and the hitting node is tied, the stiffness matrix is re-evaluated for the new configuration. Consequently, the prediction for subincrement $j+1$, $\Delta d_{j+1}$, (after the split $j$) can be computed. The prediction is followed by a sequence of corrections $\Delta d_{j+1}$, $i = 1, 2, 3, ...$ until convergence takes place. The change of the displacements in subincrement $j+1$ are denoted by $\Delta d_{j+1} = \Sigma \Delta d_{j+1}$ (summation over $i$) and consequently, the displacements are $u_k + d_j + \Delta d_{j+1}$. When the solution has converged and no nodes are penetrating any more, the solution is $d_{j+1} + \Delta d_{j+1} = d_{j+1} + \Delta d_{j+1} = D_u$. This convention is schematically illustrated in figure 4-2.

The scale factor can be calculated as demonstrated in figure 4-3 where hitting node 1 is tied to target segment 1 (in the figure indicated with the encircled 1). The penetration is calculated as discussed in chapter 2 by the determination of the gap $g$. The displacements $d_{j+1}$ of the first subincrement yield

$$d_1 = \alpha_1 (\Delta d_1) \Rightarrow \alpha_1 = \frac{g(u_k)}{g(u_k) - g(u_k + \Delta d_1)} \quad 4.8$$
where $g(u_k)$ is the gap for the configuration at the end of the previous increment. The displacements $d_{j+1}$ of the subsequent scaled subincrements yield

$$d_{j+1} = d_j + \alpha_{j+1} (Dd_{j+1}) \Rightarrow \alpha_{j+1} = \frac{g(u_k + d_j)}{g(u_k + d_j) - g(u_k + d_j + Dd_{j+1})} \tag{4.9}$$

Notice, that the gap is negative when a hitting node is penetrating. When a node does not penetrate the target body $\alpha \geq 1$. The scaling in (4.8) is linear and therefore when the movement is nonlinear this scaling is an approximation. The error of this will become larger when the nonlinearity increases (e.g. large rotations) but it can be improved by an iterative process that reduces the gap.
4.2.2 Increment splitting in the path-following methods

Now that increment splitting is considered, the relations for the prediction and the correction must be derived. This will be done for both the Riks as well as the Crisfield implementation. The first step is to compute a converged solution using the relations as introduced in chapter, (3.18) and (3.20) for the Riks method and (3.23) for the Crisfield method. When the fictitious solution has nodes that are penetrating, the increment is split with equation (4.8). The calculation of the next subincrement is discussed for the general case of subincrement j+1 after splitting j in increment k+1:

In the Riks method only the prediction contains a term for the arclength. Therefore, one need the step length for the prediction of the next subincrement, and thus for the remaining part of the increment with steplength $\Delta \eta$. This length for the next subincrement can be defined by subtracting the norm of the reduced increment $l_{j+1}^k$ from the desired arclength $\Delta \eta$ of the increment, $\Delta \eta_{j+1} = \Delta \eta - l_{j+1}$. The displacements corresponding to the first part of the increment $d_j$ and the displacements of the prediction of the next subincrement $\Delta d_{j+1}^0$ are not parallel in general. When the angle between the two parts is large, the norm of the resulting displacements is not equal to the sum of the norms, $l_{j+1} + l_{j+1}^0 \neq l_{j+1}$. However, this matter is not very important from the practical point of view and it will be ignored.

Substituting $\Delta \eta_{j+1}$ for $\Delta \eta_{k+1}$ in equation (3.18) and the prediction for subincrement $j+1$ will be

$$
\Delta \lambda_{j+1}^0 = \frac{[Du_k] \Delta \eta_{j+1} - [Du_k] \lambda^0}{[Du_k]^T L^0}
$$

$$
\Delta \nu_{j+1}^0 = \begin{bmatrix} \Delta d_{j+1}^0 \\ \Delta \lambda_{j+1}^0 \end{bmatrix} = \begin{bmatrix} R^0 + \Delta \lambda_{j+1} \\ \Delta \lambda_{j+1} \end{bmatrix}
$$

Notice that the change of the displacement vector of the previous increment $Du_k$ is used as the normal to the auxiliary surface and not the previous subincrement. Using the foregoing part of the increment has the disadvantage that the split part can be relatively small, which can cause a turn of the direction of the process. Substituting $d_{j+1}^i$ for $Du_{k+1}^i$ in equation (3.20) and the corrector for subincrement $j+1$ is given by

$$
\Delta \lambda_{j+1}^{i+1} = \frac{d_{j+1}^i}{d_{j+1}^i L^i}
$$

$$
\Delta \nu_{j+1}^{i+1} = \begin{bmatrix} \Delta d_{j+1}^{i+1} \\ \Delta \lambda_{j+1}^{i+1} \end{bmatrix} = \begin{bmatrix} R^i + \Delta \lambda_{j+1}^{i+1} L^i \\ \Delta \lambda_{j+1}^{i+1} \end{bmatrix}
$$

where the displacements due to the load $L$ and the residuals $R$ are for the corresponding cycle but for brevity the subscript $j+1$ for the splitting is dropped.

The procedure introduced by Crisfield is changed by the introduction of the split part of the increment $d_{j+1}^i$ and the change of the displacements $\Delta d_{j+1}$. 

58
\[
\|d_j + Dd_{j+1}^i\| - \Delta \eta = \|d_j + Dd_{j+1}^i + \Delta d_{j+1}^i\| - \Delta \eta = 0
\]

Recalling equation (3.25), the quadratic equation for \(\Delta \lambda\) yields

\[
a_1 (\Delta \lambda_{j+1}^i + L) + a_2 \Delta \lambda_{j+1}^i + a_3 = 0
\]

\[
\Delta \lambda_{j+1}^i = \begin{bmatrix} \Delta d_{j+1}^i \\ \Delta \lambda_{j+1}^i \end{bmatrix} = \begin{bmatrix} R^i + \Delta \lambda_{j+1}^i L^i \\ \Delta \lambda_{j+1}^i \end{bmatrix}
\]

where

\[
a_1 = [L]^T L
\]

\[
a_2 = 2 [L]^T Dd_{j+1}^i + 2 [L]^T R^i + 2 [L]^T d_{j+1} + \Delta \eta^2
\]

\[
a_3 = [d_j]^T d_j + [Dd_{j+1}^i]^T Dd_{j+1}^i + [R^i]^T R^i - \Delta \eta^2 + 2 ([Dd_{j+1}^i]^T d_j + [Dd_{j+1}^0]^T R^i + [d_j]^T R^i)
\]

This computation is carried out at each iteration. Please note that for brevity, the subscript \(j+1\) for the splitting is dropped for the terms \(L\) and \(R\).

### 4.3 Redefinition of the auxiliary surface

In section 4.2, it was argued that at the singular point \(v^i\), \(L_{(v,v)} \neq L_{(v,v)}\). The discontinuity that causes the largest change of the tangent is the geometrical discontinuity. When e.g. the hitting nodes have the largest incremental displacements just before contacting, a major change of the direction of the tangent may occur. Because the displacements of the hitting nodes do not change after contact, the increments of the displacements elsewhere in the structure become larger to fit the desired arc length of the current step. This has the effect that the rate of change of the displacements, i.e. the tangent changes direction.

Suppose that contact just occurred for the displacements \(u_k\), thus, the first hitting node is contacting the target body at the end of the increment. Suppose furthermore, that the target body is considerably stiffer than the hitting body. Consequently, the stiffness will increase significantly for the next increment \(k+1\). This is illustrated in figure 4-4. The prediction for this increment with index \(k+1\) is given by equation (3.16)

\[
K(v_k) \Delta u_{k+1}^0 + L \Delta \lambda_{k+1}^0 = 0
\]

\[
\begin{bmatrix} m^T v_{k+1}^0 \end{bmatrix} = \begin{bmatrix} n^T \Delta u_{k+1}^0 + n_0^T \Delta \lambda_{k+1}^0 \end{bmatrix} = \Delta \eta_{k+1}
\]

and when is set that \(n_0 = 0\), the latter relation of (4.10) reduces to

\[
n^T \Delta u_{k+1}^0 = \Delta \eta_{k+1}
\]
Figure 4-4, Illustration of the singularity in the prediction when contact has just occurred

The norm of the latter equation leads to $\|n^T \Delta u_{k+1}^0\| = \Delta \eta_{k+1}$ and with the Schwartz inequality this results into

$$\|n^T\| \cdot \|\Delta u_{k+1}^0\| \geq \Delta \eta_{k+1}$$

Because $\|n\| = 1$ the step in the prediction is always larger than the intended step $\Delta \eta_{k+1}$. This can immediately be seen from

$$n^T \Delta u_{k+1}^0 = \|n^T\| \cdot \|\Delta u_{k+1}^0\| \cos \theta = \Delta \eta_{k+1}$$

When the angle $\theta$, between $n$ and $\Delta u_{k+1}^0$ is zero, the length of the prediction is equal to the intended step $\Delta \eta_{k+1}$. However, when $\cos \theta$ goes to $\pi/2$, the step length of the prediction gets unbounded. This particular situation occurs in the situation of figure 4-4 when $n$ is chosen to be the normalised displacement vector of the previous increment $Du_k/\|Du_k\|$. This problem was mentioned before in section 3.3 and it is a common problem in contact analysis. However, when the tangent of the prediction $L_{k+1}^0$ is chosen, the problem does not occur.

In the MARC-code, this prediction is modelled by substituting the normalised tangent $L_{k+1}^0/\|L_{k+1}^0\|$ for $Du_k/\|Du_k\|$ in equation (3.18) which results in a modified prediction

$$\Delta u_{k+1}^0 = R^0 + \Delta \lambda_{k+1}^0 L^0$$

with:

$$\Delta \lambda_{k+1}^0 = \frac{\Delta \eta \|L^0\|}{\|\gamma L^0\|} \left[ \gamma L^0 \right]^T R_0$$

$$\gamma = \frac{Du_k^T L^0}{\|Du_k^T L^0\|}$$

The variable $\gamma$ is introduced to assure that the direction of the tangent points in the direction of the previous step and is defined as the normalized inner product of the displacements due to the load $L^0$ and the displacements of the previous step $Du_k$. The Crisfield method does not have this problem because the step length is bounded by the quadratic equation.
4.4 Considerations about the steplength

In chapter 3, it was argued that for a smooth path the changes of the arclength are affected by both the local curvature of the path and the convergence behaviour. This gradual change of the arclength cannot be applied when singularities occur in the path due to the discontinuity of the stiffness. A step \( \Delta \eta_{k+1} \), based on the behaviour of the path before contacting may cause difficulties after contact because it may be too large. Even when the process converges, the step can be such that details of the path are lost. Therefore, a redefinition of the arclength is required at contact. The issue is now how to define such an arclength.

This arclength can never be found automatically. It will be necessary to apply a method of trial and error. As mentioned, the step can be such that details of the part of the path between the two subsequent points are lost. Therefore, it turns out to be convenient to introduce a tolerance relative to the surface, which limits the arclength \( \Delta \eta_{k+1} \). This tolerance is defined in such a way that the hitting nodes do not penetrate more than a particular distance (defined by the tolerance) and it is used to scale the fictitious solution before the increment is split. In section 2.4 a tolerance was defined relative to the element measurements, but this new tolerance concerns the penetration and it is determined by the geometry of the contacting bodies and the configuration of the problem. Because it is dependent of the configuration, e.g. the stiffness, it can hardly be defined automatically. This choice is thus left to the user. The corresponding step length can be determined by scaling the fictitious solution \( Dd_0 \) (with penetrations). The reduction of the solution must be such, that the node for which the penetration is the largest is scaled to a magnitude of the penetration that is equal to the newly defined tolerance. This scaled increment can be used as the first fictitious solution which should be split. The arclength according to this scaled increment can be used as the starting value for \( \Delta \eta \) in the following steps.

The use of the tolerance to reduce the penetration can be limited to the first time contact takes place. However, this convention has the disadvantage that the subsequent steps may become larger and larger later on in the process. For example, it is possible that the hitting body comes in contact with the target body for the second time, so that it becomes necessary to use the tolerance again. It is therefore, preferable to use the tolerance for each time nodes are penetrating and scale the increment accordingly, so that the maximum penetration remains within the desired range. Notice, that in this case the increment is scaled several times. Firstly, for the maximum penetration, and subsequently, for the splitting of the increment. After the first scaling an iteration to get a converged solution is useful but not strictly necessary. The scaling can then be done in the same way as discussed in section 4.2.

4.5 Problems of large residuals in the implementations

The residuals during the iterative process in contact analysis can become very large as compared to the residuals in problems without contact. We have seen that there are several sources that can cause residuals in contact analysis. For example, contact tolerances on the target surface and separation forces. Especially, when the norm of the displacement vector due to the residuals \( \|R\| \) becomes of the order of the arclength \( \Delta \eta \), problems occur. These problems can be solved by requiring smaller convergence tolerances. In contact analysis this will not always solve the problem. For example, when the residuals are caused by a configuration of the structure which is not longer in equilibrium and cannot be solved
with a quasi-static solution procedure, the mentioned measure will not solve the problem. This change from statics to dynamics occurs when a part of the structure is losing contact and a dynamic jump to its original equilibrium configuration is about to take place. Both implementations that are discussed here cannot overcome this problem because at this point transient rather than static methods are needed to continue the calculations.

There is yet another difficulty that needs consideration which is particular to the Crisfield implementation of the path-following method. This method has problems with large residuals as was already mentioned in section 3.4. With large residual forces in the model, a situation of no real roots for the quadratic equation is possible. To solve this difficulty, Crisfield suggested the reduction of the step $\Delta \eta_{k+1}$. In contact analysis, separation forces can also cause large residuals. In that case, the problem cannot be solved by reducing the arclength or tightening of the convergence criteria. Figure 4-5 shows the problem of the separation of a hitting and a target node, node 1 and node 2, respectively. Here the target nodes must be returned to the equilibrium configuration when separation has occurred. From the figure it can be seen that when the displacement gap is too large relative to the step length $\Delta \eta_{k+1}$, no intersection between the equilibrium path and the circle in the figure can be found. The solution, in this case, may be found in either reducing the separation force or by increasing the arclength $\Delta \eta$. 

Figure 4-5 Demonstration of problem with large residuals for Crisfield’s implementation of the path-following method
Chapter 5

Conclusions

5.1 Conclusions

In this study the difficulties were discussed that occur when the path-following method is used for contact analysis. The discussion about contact in FEM is in accordance with the way MARC modelled contact in the code. The most important aspects of contact in the MARC-code are, firstly, the reduction of the degrees of freedom due to contact, and secondly, the discontinuous points in the equilibrium path.

From these considerations the following conclusions can be drawn. The equation that prescribes the auxiliary surface is augmented to the underdetermined system of equilibrium equations and should be properly defined to prevent singularities. The selection of the extra equation should be based on the simplicity of the formulation. The quadratical equation Crisfield proposed to augment the system may introduce an additional singularity and therefore, when the nonlinearity increases, this method is less suitable to solve nonlinear problems. Riks implementation of the path-following method is based on a linearised augmented system and therefore does not have the difficulty of Crisfields implementation. For organisational reasons, the implementation in the MARC-code is slightly different from the implementation proposed by Riks. These differences make the method less robust under highly nonlinear circumstances.

The path-following method has been successfully used to solve contact problems with MARC. When the system of equilibrium equations is transformed to the full system, the reduction of the number of the degrees of freedom due to the tyings has no impact on the formulation of the relations for the path-following method. Contact imposes discontinuity on the tangent to the solution curve, and therefore, on the direction in which the solution proceeds. To deal with discontinuity of the tangent, the analysis should be stopped when a node is hitting the target surface and at this point the stiffness must be re-evaluated before resuming the computation. This break in the analysis is preferably obtained by splitting of the increment in subincrements, such that the intended step length of the increment is preserved. The change in direction at discontinuity may lead to a singular condition when the normal to the auxiliary surface in the extra equation is based on the previous part of the path. Best choice in this case is to use the tangent of the solution curve after the split. This split in the analysis can also be used to redefine the arclength for the analysis after contact. The new arclength can be based on the maximum allowable penetration before the fictitious solution is scaled. This fictitious solution is a solution at which penetrating nodes are allowed. It should finally be noted that contact analysis, in general, causes larger residual forces then analyses without contact.

5.2 Recommendations

The recommendations can be divided in the recommendations for MARC code and the recommendations for the Faculty of Aerospace Engineering of Delft University of Technology. Firstly the recommendations for MARC are
• Change, for the implementation of Riks' method, the definition of the normal to the auxiliary surface during the iterative process and neglect the residuals in the prediction. This according to the discussion in section 3.4.
• Study the possibilities to solve problems with large residuals e.g. due to loss of contact when the quasi-static formulation of the problem is no longer justified.
• Study the use of the path-following method according to Riks for contact analysis with sliding, and when this is solved, the method according to Crisfield can be removed to make the options of auto increment in MARC easier to understand.
• Expand the possibilities of the path-following method for bifurcation analysis. This possibilities can be based on the method discussed by Riks.

Secondly, the recommendations for the Faculty of Aerospace Engineering of Delft University of Technology are:

• Study the different implementations of contact in an implicit finite element method, e.g. B2000.
• Study the discussed problems and measurements for the use of the path-following method for contact analysis for the Lagrange and the penalty method.
References

Appendix A

A.1 Geometrical interpretation of the path-following method

It is possible to give a geometrical interpretation of the derived arclength method but it will be quite abstract because the $N$ displacements will be plotted in the norm of the displacements. A step in the iterative process for the iterate $v_{k+1}^{i+1}$ is considered.

A.1.1 Geometrical interpretation of the linearizations

The curve $C$, given by the system $f(v) = 0$ is augmented by $f_{N+1} = h(v) - \eta = 0$. Both are drawn as thick lines in figure 5-2. The starting point of the process is an approximated solution $v_{k+1}^i$ which is not on the solution curve because convergence has not taken place yet. The curve $f(v_{k+1}^i) = r$ is the dashed line through the approximation $v_{k+1}^i$. When these non-linear curves are linearised it results in two straight lines through the previous iterate $v_{k+1}^i$, one in the direction tangent to curve $f(v_{k+1}^i) = r$, given by $v'$, and one tangent to $f_{N+1}$. Recalling equation (3.9) it follows that

\[
v^{i+1} = v^i + \Delta v^{i+1} = v^i + \begin{bmatrix} R^i \\ 0 \end{bmatrix} + \Delta \lambda^{i+1} \begin{bmatrix} L^i \\ 1 \end{bmatrix}
\]

$S_1$: tangent to the approximated auxiliary surface
$S_2$: tangent to the approximated solution, moved by the residual $R$

Figure 5-1 Geometrical interpretation of the corrector process
where $\mathbf{R}$ and $\mathbf{L}$ are the displacements due to the residual forces $\mathbf{r}$ and the total applied load $\mathbf{l}$, respectively. This vector $(\mathbf{L}, \mathbf{l})$ is the tangent to curve $C$, in $v_{k+1}^{i}$ and this tangent is scaled by $\Delta \lambda_n$ to get the iterate $v_{k+1}^{i+1}$. Consequently, the iterate $v_{k+1}^{i+1}$ is the intersection of the linearised augmented equation and the tangent through the point $v_{k+1}^{i} + \mathbf{R}$ (solution 2 in the figure).

### A.1.2 Further considerations

An assumption that is frequently made in analyses where the non linearity is small is not updating the tangent to the approximated curve of the equilibrium solution. Not updating the tangent during the iteration may result in the desired solution of the problem when the steps are not too big. This leads in general to an increasing number of iterations. Much more important is the advantage of not updating the stiffness matrix each iteration which speeds up the process and makes the operation less costly. The method with this assumption is called the modified Newton-Rapson.

Furthermore, the situation of no real roots in the implementation of Crisfield is discussed. For simplicity the situation is shown for an arbitrary equilibrium curve in the plane of the displacements $u_1$ and $u_2$ which are arbitrary chosen from the $N$ degrees of freedom. Assume that the equilibrium path in the $u_1$-$u_2$ plane looks like figure 5-2. The situation is considered for two different auxiliary surfaces where the difference is the magnitude of the pseudo arclength, and thus the radius in figure 5-2. The situation with a larger radius is plotted with continuous lines while the situation with the smaller radius is plotted with dashed lines and the variables of the latter are denoted with an asterisk. The point $A$ and $A^*$ are the predictions $v_{k+1}^{0}$ for respectively the larger and the smaller pseudo arclength. The second step for both situations is to compute the intersection between the cylindrical constraint (the circle in the figure) and the tangent of the path through point $A$, respectively $A^*$, moved by the vector of displacements due to the

![Diagram](image-url)

*Figure 5-2: Geometrical interpretation of the roots of the cylindrical constraint*
residuals $\mathbf{R}$, respectively $\mathbf{R}^*$. In the figure the situation is plotted that no solutions exists for the case with the large pseudo arclength and thus a large step. When the pseudo arclength is reduced (a smaller step) solutions possibly exists (the squares in the figure). This is in general true but later it turns out that in combination with mechanical contact a reduction of the steplength, recommended by Crisfield, does not solve the problem of imaginary roots.

A.2 Path-following procedures

A.2.1 Procedure of Riks’ implementation

The continuation process of a linear prediction and correction according to Riks using the equations (3.18) - (3.20) will be:

4. determine the desired arclength $\Delta \eta_{k+1}$ by using the norm of the displacements of the previous step $\|\mathbf{D}u_k\|

5. determine for the configuration $u_k$ the stiffness matrix $\mathbf{K}$ and the residual vector $\mathbf{r} = f(u_k)$

6. compute the inverse $\mathbf{K}^{-1}$

7. compute the displacements due to the residual vector $\mathbf{r}$, $\mathbf{R} = -\mathbf{K}^{-1}\mathbf{r}$, and the displacements due to the load vector $\mathbf{l}$, $\mathbf{L} = -\mathbf{K}^{-1}\mathbf{l}$

8. compute the prediction

$$\Delta \lambda_{k+1}^0 = \frac{[\mathbf{D}u_k][s_{k+1} - [\mathbf{D}u_k]^T\mathbf{R}^0}{[\mathbf{D}u_k]^T\mathbf{L}^0}$$

$$\Delta v_{k+1}^0 = \begin{bmatrix} \Delta u_{k+1} \\ \Delta \lambda_{k+1}^0 \end{bmatrix} = \begin{bmatrix} \mathbf{R}^0 + \Delta \lambda_{k+1}^0 \mathbf{L}^0 \\ \Delta \lambda_{k+1}^0 \end{bmatrix}$$

$$v_{k+1}^0 = v_k + \Delta v_{k+1}^0$$

9. determine for the configuration $u_k^i$ the stiffness matrix $\mathbf{K}$ and the residual vector $\mathbf{r} = f(u_k^i)$

10. compute the inverse $\mathbf{K}^{-1}$

11. compute the displacements due to the residual vector $\mathbf{r}$, $\mathbf{R} = -\mathbf{K}^{-1}\mathbf{r}$, and the displacements due to the load vector $\mathbf{l}$, $\mathbf{L} = -\mathbf{K}^{-1}\mathbf{l}$

12. calculate the loadstep with

$$\Delta \lambda_{k+1}^{i+1} = \frac{[\mathbf{D}u_k^i]^T\mathbf{R}^i}{[\mathbf{D}u_k^i]^T\mathbf{L}^i}$$

$$\Delta v_{k+1}^{i+1} = \begin{bmatrix} \mathbf{R}^i + \Delta \lambda_{k+1}^{i+1} \mathbf{L}^i \\ \Delta \lambda_{k+1}^{i+1} \end{bmatrix}$$

13. correct the approximation $v_{k+1}^i$ with,
\[ v_{k+1}^{i+1} = v_{k+1}^{0} + \sum_{n=1}^{i+1} \Delta v_{k+1}^{n} = v_{k+1}^{0} + Dv_{k+1}^{i+1} \]

14. when the solution is converged, \( v_{k+1}^{i+1} = v_{k+1}^{+} \) and goto step 1, next increment. when the solution is not converged goto step 6.
Notice that step 6,7,8 are equal to step 2,3,4

**A.2.2 Procedure of Crisfield's implementation**

The continuation process of the implementation according to Crisfield will be:
1. determine the desired arclength \( \Delta \eta_{k+1} \) by using the norm of the displacements of the previous step \( \|Du_k\| \)
2. determine for the configuration \( u_k^i \) the stiffness matrix \( K \) and the residual vector \( r = f(u_k^i) \)
3. compute the inverse \( K^{-1} \)
4. compute the displacements due to the residual vector \( R = -K^{-1}r \), and the displacements due to the load vector \( I, L = -K^{-1}I \)
5. compute
\[
\begin{align*}
a_1 &= [L^i]^T L^i \\
a_2 &= 2[L^i]^T Du_{k+1}^i + 2[L^i]^T R^i \\
a_3 &= [Du_{k+1}^i]^T Du_{k+1}^i + 2[R^i]^T R^i + [R^i]^T R^i - (\Delta \eta)^2
\end{align*}
\]
6. calculate the two roots of
\[
a_1 (\Delta \lambda_{k+1}^{i+1})^2 + a_2 \Delta \lambda_{k+1}^{i+1} + a_3 = 0
\]
7. choose the root in the direction of interest for the prediction by \( [Du_k]^T Du_{k+1}^0 > 0 \)
8. choose the root in the direction of interest for the prediction by \( [Du_k^i]^T Du_{k+1}^{i+1} > 0 \)
9. calculate
\[
\Delta v_{k+1}^{i+1} = \begin{bmatrix} R^i + \Delta \lambda_{k+1}^{i+1} L^i \\ \Delta \lambda_{k+1}^{i+1} \end{bmatrix}
\]
10. compute,
\[
v_{k+1}^{i+1} = v_{k+1}^{0} + \sum_{n=1}^{i+1} \Delta v_{k+1}^{n} = v_{k+1}^{0} + Dv_{k+1}^{i+1}
\]
11. when the solution is converged, \( v_{k+1}^{i+1} = v_{k+1}^{+} \) and goto step 1, next increment. when the solution is not converged goto step 2.
Appendix B

B.1 Demonstration of problems in contact

Some of the problems that are mentioned in chapter 4 about path-following methods for contact analysis can be shown with a simple two bar problem. Two bars will be considered which will contact in the second increment. The geometry is as presented in figure A-1. The subscribed displacement for node 1 and thus the total applied load is 0.3. This will be applied by the Riks implementation. Two steps with constant step length will be considered. The first step is chosen to be the distance between the two bars. At the end of the first increment, node 2 must be tied to node 3. Consequently, the second step of the process is the step of interest.

Assuming that the bars are linear for the considered problem and the stiffness of the two bars are equal the matrix equation will be

\[
Ku = \begin{bmatrix}
  k & -k & 0 & 0 \\
  -k & k & 0 & 0 \\
  0 & 0 & k & -k \\
  0 & 0 & -k & k \\
\end{bmatrix}
\begin{bmatrix}
  u_1 \\
  u_2 \\
  u_3 \\
  u_4 \\
\end{bmatrix}
= F
\]

The first step is just the rigid body translation of the first bar (between nodes 1 and 2) with magnitude 0.1. The displacements and the arc length for the first step will be

\[
Du_1^T = [0.1, 0.1, 0, 0] \quad \Rightarrow \quad \Delta \eta_1 = (Du_1^T Du_1)^{1/2} = \sqrt{0.02}
\]

hitting node 2 is contacting node 3 and, consequently, the displacements of the two nodes must be equal in the second increment. The tying relation \( u_2 = u_3 + 0.1 \) results in a reduced stiffness matrix \( \bar{K} \)

![Figure B-1 Illustration of the configuration for the two bar contact](image)

*Figure B-1 Illustration of the configuration for the two bar contact*
\[
\tilde{\mathbf{K}} = \begin{bmatrix}
k & -k & 0 \\
-k & 2k & -k \\
0 & -k & k
\end{bmatrix} \quad \tilde{\mathbf{u}} = \begin{bmatrix}
u_1 \\
u_3 \\
u_4
\end{bmatrix} \quad \tilde{\mathbf{F}} = \begin{bmatrix}
F_1 + 0.1k \\
F_3 + F_3 - 0.1k \\
F_4
\end{bmatrix}
\]

where \( \tilde{\mathbf{u}} \) is the reduced displacement vector and \( \tilde{\mathbf{F}} \) is the reduced force vector. The subscribed displacement for node 1 can be imposed by

\[
\begin{bmatrix}
1 & 0 & 0 \\
-k & 2k & -k \\
0 & -k & k
\end{bmatrix} \begin{bmatrix}
u_1 \\
u_3 \\
u_4
\end{bmatrix} = \begin{bmatrix} 0.3 \\
-0.1k \\
F_4
\end{bmatrix} \quad \Rightarrow \quad u_3 = 0.1 \quad L = \begin{bmatrix} 0.3 \\
0.1 \\
0
\end{bmatrix}
\]

where \( L \) are the total described displacements. The residuals are 0 because the bars are linear and therefore the input for the incremental method according to Ramm are known. The prediction for the second step of the analysis is

\[
\Delta \lambda_2^0 = \frac{\|D\tilde{u}_1\|\Delta n_2 - [D\tilde{u}_1]^T R^0}{[D\tilde{u}_1]^T L^0} = \frac{\|\Delta \tilde{u}_1\| s_2}{\Delta u_1^T L} = 0.02 \quad 0.04 = 0.5
\]

\[\Delta u_2 = 0.5 \cdot [0.3, 0.1, 0.1, 0]^T = [0.15, 0.05, 0.05, 0]^T
\]

The total displacement of the first node is 0.25, which is bigger than the 0.2 without contact. This difference is due to the arclength which is defined to be constant because the displacements of the second and the third node are smaller. The intended step length is the square root of 0.02 but when the arclength is calculated this gives

\[s_1 = (\Delta u_2^T \Delta u_2)^{1/2} = \sqrt{0.0275}
\]

This value for the arclength is not equal to the intended step and this is due to the inner product of the normal to the auxiliary surface, which is the previous step, with the tangent. Notice that the degrees of freedom are independent and thus the vectors may not be parallel unless the displacements are geometrically in line. Applying the modified method as presented in section 4.3 gives

\[
\Delta \lambda_2^0 = \frac{s\|L_0\| - [\gamma L_0^0]^T R_0}{[\gamma L_0^0]^T L_0^0} = \frac{\sqrt{0.02} \cdot \sqrt{0.11}}{0.11} \quad \Rightarrow \quad \gamma = \frac{\Delta u_1^T L_0^0}{\|\Delta u_1^T L_0^0\|} = 1
\]

Substitution gives the displacements of the second step and the resulting arclength is 0.02. The calculation without the modification does not fail but there are problems where the calculation indeed fails. The modification is then necessary to solve the problem.

---

72