Efficient Simulation and Process Mechanics

of

Incremental Sheet Forming

A.M.H. Hadoush

Committee members:

Chairman:	
Prof. dr. ir. F. Eising	University of Twente
Promoter:	
Prof. dr. ir. J. Huétink	University of Twente
Assistant promoter:	
Dr. ir. A.H. van den Boogaard	University of Twente
Members:	
Prof. dr. ir. R. Akkerman	University of Twente
Prof. dr.-ing. S. Reese	RWTH Aachen
Dr. ir. W.C. Emmens	Corus Research, Development and Technology
Dr. A.M. Habraken	University of Liège
Dr. ir. R. Hagmeijer	University of Twente

ISBN 978-90-365-3052-1

DOI 10.3990/1.9789036530521

1st Printing June 2010

Keywords: Efficient implicit time integration, Adaptive remeshing, incremental sheet forming, continuous bending-under-tension

This thesis was prepared with LaTeX by the author and printed by Ipskamp, from an electronic document.

EFFICIENT SIMULATION AND PROCESS MECHANICS

OF

INCREMENTAL SHEET FORMING

DISSERTATION

to obtain
the degree of doctor at the University of Twente,
on the authority of the rector magnificus,
prof. dr. H. Brinksma,
on account of the decision of the graduation committee,
to be publicly defended
on Thursday, 1 July 2010 at 13.15 hrs.

by

Ashraf Moh’d Hasan Hadoush

born on 5 February 1980
in Mecca, Saudi Arabia

This thesis has been approved by:

Prof. dr. ir. J. Huétink (promoter)
Dr. ir. A.H. van den Boogaard (ass. promoter)

Contents

Summary

Single Point Incremental Forming (SPIF) is a displacement controlled process performed on a CNC machine. A clamped blank is incrementally deformed by the movement of a small-sized tool that follows a prescribed lengthy tool path. The strain achieved by the SPIF process is higher than the strain achieved by classical forming processes e.g. deep drawing. This motivated many researchers for the last two decades studying the process mechanics and still a definite explanation is missing. The finite element method is a powerful tool in studying the forming processes. Compared to e.g. deep drawing, the FE model for SPIF is very simple. However, simulation of the process is a challenging task because of the enormous computing time as a result of performing thousands of load increments on a relatively fine FE model. This limits the use of the finite element method to simple academic cases that already require weeks of computing time. The focus of this thesis is to efficiently use the implicit time integration method in order to drastically reduce the required computing time for incremental forming simulation.

Because of the localised plastic deformation, the part of the FE mesh that is in the vicinity of the tool experiences a strong nonlinearity. The strong nonlinearity is a combination of the material and geometrical nonlinearities. The rest of the FE mesh that models the elastically deforming part of the blank experiences only a weak geometrical nonlinearity. Using the standard Newton method is required because of the strong nonlinearities in the set of equations, but it is an expensive update procedure and it is inefficiently used for the large elastically deforming part. Therefore, it becomes necessary to have a different treatment that is accurate and computationally efficient for different parts of the FE mesh. The fully Newton nonlinear treatment is used for the localised plastic deformation. The rest of the FE mesh that is elastically deforming is treated by a pseudo-linear approach. The pseudo-linear treatment applies a nonlinear geometrical and material update for the tangent stiffness matrix and the internal force vector only once every increment or number of increments. Within the increment(s), the tangent stiffness matrix is reused, as in the modified Newton method. The internal force vector is linearly updated by the multiplication of the tangent stiffness matrix and the incremental displacement vector. It is a relatively cheap update procedure compared to the Newton method.

The partitioning of the FE mesh into domains with different update strategies (iteratively, incrementally and multi-incrementally) can be done by several indicators. Here, three indicators are developed for incremental sheet forming in order to generically classify these domains. These indicators are the current tool location, plastic deformation in the previous load increment and the shape change in the previous load increment. The tool indicator and the plastic history indicator are suitable to classify the FE mesh into the iterative and the incremental update strategies. The geometrical indicator is used to determine the needs of updating a multi-incremental domain.

An analytical formula is derived for SPEED which measures the performance of the efficient implicit method in speeding up the standard implicit simulation of an incremental forming process. It is defined as the CPU time cost of one Newton increment to the cost of one increment of the efficient implicit method. SPEED depends on several factors: the number of the iterations used per increment, the used update strategies, the size of the domains and the cost of major parts of the Newton iteration (building the system of equations, solving it and updating the stresses). For a simple material model and finite element type, the efficient implicit method can accelerate a SPIF simulation with negligible iterative zone size and negligible solving cost by a factor approximately equal to the number of the iterations used per increment. Furthermore, the advantage of adaptive refinement is combined with the efficient implicit method resulting in an additional acceleration of the implicit simulation of a SPIF process.

In addition, this thesis presents a fundamental study on a particular aspect of the process mechanics involved in the SPIF process. The study is carried out on the continuous bending under tension (CBT) process. It has the advantage of reducing the 3-dimensional complex bending in the SPIF process to a merely 2-dimensional case. It is shown that combined bending and tension can stabilize the deformation of a strip to a high level of strain. An increase of the force is required to introduce additional stable deformation. This condition requires that the averaged tangent stiffness has to be larger than the averaged stress. The presence of compressive stress reduces the average stress while the elastic fibers increase the average tangent stiffness of the cross section. Bending introduces both the compressed fibers and the elastically loaded fibers. A further analysis is carried out on the achieved cyclic force–displacement curve of the CBT test. The cycle consists of two parts: steady and transient. The part having a steady level of force represents the deformation of the strip governed by significant curvature change of the strip because of bending. The transient increase of the force results from the deformation of the strip by increasing the tension force with no significant change in strip curvature.

Samenvatting

“Single Point Incremental Forming” (SPIF) is een verplaatsing gestuurd proces dat uitgevoerd wordt op een CNC machine. Een ingeklemde plaat wordt incrementeel vervormd door de beweging van een klein gereedschap dat een voorgeschreven pad volgt. De rekken die bij het SPIF proces behaald worden zijn hoger dan de rekken die bij reguliere omvorm processen zoals dieptrekken behaald worden. Gedurende de afgelopen twee decennia motiveerde dit veel onderzoekers om de achterliggende procesmechanica te bestuderen, hoewel een sluitende verklaring nog steeds ontbreekt. De eindige-elementenmethode is een krachtig hulpmiddel in het bestuderen van omvormprocessen. Vergeleken met bijvoorbeeld het dieptrek proces, is een eindig-elementenmodel voor het SPIF proces relatief eenvoudig. Echter, simulatie van het proces is een uitdagende taak vanwege de benodigde lange rekentijden. Als gevolg van het uitvoeren van duizenden belastingincrementen op een relatief fijn eindige-elementenmodel neemt de rekentijd snel toe. Dit beperkt het gebruik van de eindige-elementenmethode tot enkele vereenvoudigde academische gevallen die overigens ook al weken aan rekentijd vereisen. De focus van dit werk is het efficiënt toepassen van de impliciete tijdintegratiemethode om de vereiste rekentijd voor simulaties van het incrementeel omvormproces te verkorten.

Vanwege de lokale plastische deformatie ondervindt een deel van het eindige-elementennet in de buurt van het gereedschap een sterke niet-lineariteit. Deze niet-lineariteit is een combinatie van materiaal- en geometrische niet-lineariteiten. Het overige deel van het eindige-elementennet dat het elastisch vervormde deel van de plaat modelleert, ondervindt alleen een zwakke geometrische niet-lineariteit. Het gebruik van de standaard Newton methode is noodzakelijk vanwege de sterke niet-lineariteit in de set van vergelijkingen. Het is echter een kostbare procedure en bovendien inefficiënt in gebruik voor het elastisch vervormende deel. Een alternatieve aanpak is daarom noodzakelijk die zowel nauwkeurig als efficiënt is met betrekking tot de rekentijd voor verschillende delen van het eindige-elementennet.

De volledige niet-lineaire Newton benadering is gebruikt voor de lokale plastische deformatie. Het resterende deel van het eindige-elementennet, wat elastisch vervormd wordt, is behandeld met behulp van een pseudo niet-lineaire benadering. De pseudo niet-lineaire benadering past slechts éénmaal per increment, of over een aantal incrementen, een niet-lineaire geometrische en materiaal correctie toe voor de tangentiële stijfheidsmatrix en de interne krachtvector. De tangentiële stijfheidsmatrix wordt hergebruikt in het increment zoals ook toegepast wordt in de gemodificeerde Newton methode. De interne krachtvector wordt lineair geupdate door de vermenigvuldiging van de tangentiële stijfheidsmatrix en de incrementele verplaatsingsvector. Dit resulteert in een relatief efficiënte correctie procedure in vergelijking met de Newton methode.

De opdeling van het eindige-elementennet in domeinen met verschillende correctie strategieën (iteratief, incrementeel en multi-incrementeel) kan gedaan worden met behulp van verschillende indicatoren. Drie indicatoren zijn ontwikkeld voor incrementeel plaatomvormen om de domeinen in algemene zin te definiëren. Deze indicatoren zijn gebaseerd op de huidige locatie van het gereedschap, de plastische deformatie in het vorige belastingincrement en de vormverandering in het vorige belastingincrement. De positie-indicator en de plastische-geschiedenis-indicator zijn geschikt om het eindige-elementennet in de iteratieve en incrementele update strategie te classificeren. De geometrische indicator wordt gebruikt om te bepalen of een correctie in een multi-incrementeel domein noodzakelijk is.

Een analytische formule is ontwikkeld voor SPEED wat de prestatie van de efficiënte impliciete methode meet in het versnellen van de standaard impliciete simulatie van een incrementeel omvormproces. Het is gedefinieerd als de verhouding van de CPU tijd van een Newton increment en een increment van de efficiënte impliciete methode. SPEED is afhankelijk van verschillende factoren: aantal benodigde iteraties per increment, toegepaste correctie strategie, domeingrootte en kosten van de hoofdonderdelen van de Newton iteratie (opstellen van het stelsel van vergelijkingen, oplossen en bepaling van de spanningen). Voor een simpel materiaalmodel en eindige-elementtype kan de efficiënte impliciete methode een SPIF simulatie, met een verwaarloosbare iteratieve zone en verwaarloosbare oplossingskosten, versnellen met een factor ongeveer gelijk aan het aantal te gebruiken iteraties per increment. Bovendien wordt adaptieve elementennet verfijning gecombineerd met de efficiënte impliciete methode, resulterend in een extra versnelling van de impliciete simulatie van het SPIF proces.

Daarnaast wordt in dit proefschrift een fundamentele studie gepresenteerd van een specifiek aspect van de procesmechanica die plaatsvindt in het SPIF proces. Deze studie is uitgevoerd op een proces waarbij continu gebogen wordt onder trekbelasting (Continuous Bending under Tension, CBT). Dit heeft het voordeel dat het 3-dimensionale complexe buiggedrag in het SPIF proces gereduceerd wordt tot een 2-dimensionaal probleem. Het is aangetoond dat het gecombineerd buigen en trekken de deformatie van een strip tot hoge rekniveaus kan stabiliseren. Voor een constante buigradius is een toename van de kracht noodzakelijk om een stabiele deformatie te introduceren. Deze conditie vereist dat de gemiddelde tangentiële stijfheid groter moet zijn dan de gemiddelde spanning. De aanwezigheid van drukspanningen vermindert de gemiddelde spanning, terwijl de elasticiteit de gemiddelde tangentiële stijfheid van de dwarsdoorsnede doet toenemen. Buiging introduceert zowel vezels belast op druk als elastisch belaste vezels. Een verdere analyse is uitgevoerd op de behaalde cyclische kracht-verplaatsingscurve van de CBT test. De cyclus bestaat uit twee delen, respectievelijk met een stabiel krachtniveau en een krachtpiek. Het deel dat een stabiel krachtniveau ondergaat representeert de deformatie van de strip veroorzaakt door een significante verandering van de kromming van de strip als gevolg van buiging. De transiente toename van de kracht resulteert uit de deformatie van de strip als gevolg van de trekkracht toename zonder significante verandering van de buigradius.

Acknowledgments

Ya Allah, I am finally writing these words after four years of work. I do not think that PhD work is a one-man job, of course; the PhD student is simply the main driving force in the project. Still, we do not travel just by means of an engine. Therefore, I would like to express my acknowledgments and gratitude to people who have supported me during this journey.

The first person I had to contact was Prof. J. Huétink, who responded to my application by telling me they had already found another candidate, oops! However, he went on to say that he was under the impression I may be a candidate for another project. Han, many thanks for accepting me and allowing me to pursue my PhD in your group and for offering me this opportunity to learn more about applied mechanics. Thank you, Han, for the guidance and enthusiastic support for these years.

I would like to express my gratitude to my supervisor Dr. ir. A.H. van den Boogaard. Ton, you gave me full support and the chance to do this research my way. You kept a close eye on monitoring my overall performance and stopped me from blindly going in the wrong direction. You encouraged me and challenged me. Also, I had a nice time engaging in discussions with you, and these formed a valuable experience. I appreciate the efforts you made with me to present my work as it is now. Furthermore, you supported my future career. I owe you, Ton, a special thanks.

This research was carried out under the project number MC1.0 5227 in the framework of the Research Program of the Materials innovation institute M2i (www.m2i.nl), the former Netherlands Institute for Metals Research. Therefore, I would like to acknowledge M2i for granting this project.

I would like to thank Eisso Atzema (Corus RD & T) for the fruitful discussions during the regular meetings. I am grateful to Robertt A. Fontes Valente and Ricardo Alves de Sousa for giving me the chance to learn more about their solid-shell element. I would like to thank my master's student Remi Boon for his effort.

I am grateful to the reading committee comprising Han Huétink, Ton van den Boogaard, Johan Hol and Wilko Emmens who read the complete manuscript or parts of it and helped me to improve its content. Additionally, I would like to thank Vivien Cook for proof-reading the English and Jan Harmen Wiebenga for his Dutch translation of the summary.

Dieka group has a nice and friendly atmosphere, and many thanks go to Bert Geijslaers, Harm Wisselink, Timo Meinders, Nico van Vliet, Herman van Corbach and Tanja Gerrits for their time and support. Special thanks to Debbie Vrieze-Zimmerman for helping me and translating many Dutch administrative letters. Thanks to all colleagues in the applied mechanics department for the nice time we spent together; I would like to mention Didem Akcay, Semih Perdahcioglu, Pawel Owcarek, Wissam Assaad, Muhammad Niazi, Emre

Dikmen, Wouter Quak, Mahmoud Ravanan, Alejandro Martinezlopez and my office mates: Maarten van Riel, Srihari Kurikuri, Johan Hol and Jan Harmen Wiebenga.

Life in Enschede would have been difficult without my friends Abdulsalam, Mohammed Morsi, Rabah, Sameh, Wissam, Mohammed Khatib, Ala'a, Hamdi, Yousef, Ahmad, Wael, Dlovan and UT-Moselm brothers. I highly appreciate your friendship and thanks for cheering me up. I shall never forget the lovely hospitality of Hajj Omar and his wife Hajja Basma and the way they treated me as a family member. Also, I would like to extend thanks to my friends across the border, especially Mohammed Aljedi, Mohammed Shadi, Mohammed Altous, Majdi and Anas for keeping this friendship alive.

I dedicate this thesis to my parents. Language is lost for words to thank you for raising me well, doing everything you can for me, accepting my choices, giving tireless support and believing that I can achieve my goals. Rest in peace Father. Mother, your prayers lighten my way. My brothers Qasem, Ziad, Nezar, Ayman and Hikmat, my beloved sister Mai and her husband Ahmad deserve a big thanks for their support, confidence in me and for showing interest in what I do.

Last but not least, I would like to thank my fiancée Roba for her love, for being patient and for granting me a great deal of time to finish my PhD.

1. Introduction

In ancient times, if you were a knight looking to buy new chest armor you would search for a skilled blacksmith. Our skilled blacksmith used his hammer, among many simple tools, and he shaped an initially flat sheet into chest armor. Focusing on the procedure of producing the armor, it can be imagined that the blacksmith will start by warming the sheet then start hammering it; then he may use a rod with a round tip to create a proper curvature that matches the chest shape of the knight. If the knight can afford more coins perhaps the blacksmith will offer more fancy details and create a unique chest armor. In this case, the blacksmith definitely will leave a remarkable fingerprint so everybody will know who is the father of this unique piece. Sadly, neither the blacksmith nor the knight were interested in simulating the incremental forming of the sheet into a chest armor using simple tools or in studying the fundamental mechanics of the process.

1.1 Incremental sheet forming

Incremental forming is a common characteristic of several processes like ring rolling, spinning and asymmetric incremental sheet forming AISF. In these processes, a forming tool deforms a workpiece to the required geometry by a sequence of small and localized plastic deformation. Regardless of the size of the tool, the forming tool has a small contact area with the workpiece. During the process, the contact area (forming tool) travels all over the workpiece several times in loops or revolutions. Within the loop, a portion of the workpiece deforms plastically for a small time interval compared to the total process time. After each loop, the initial geometry is gradually changed toward the desired final geometry.

The process time becomes even longer when the small contact area changes from line-like (rolling) to point-like (asymmetric incremental sheet forming). Asymmetric incremental sheet forming appears in several configurations. The simplest is single point incremental forming (SPIF) where a clamped sheet is deformed by a small spherical shaped tool mounted on a CNC machine (Iseki *et al.*, 1989). The basic idea was introduced by Mason (1978). Two point incremental forming (TPIF) has the same configuration as SPIF but it uses a partial or full die to produce more diffcult details (Matsubara, 1994). Kinematic incremental sheet forming (KISF) uses another moving forming tool instead of the fixed die in TPIF (Meier *et al.*, 2007; Maidagan *et al.*, 2007).

It is known from the literature that the AISF process is favourable for prototyping and small batch production. It is a very flexible process; changing the followed tool path results in producing new geometry (product). Products that were successfully produced by AISF include a headlight (Jeswiet and Hagan, 2001), a stiffening brace (Hirt *et al.*, 2005), an ankle support (Ambrogio *et al.*, 2005a) and a cranial implant (Duflou *et al.*, 2005). An extensive overview of the asymmetric incremental sheet forming process has been presented by Jeswiet *et al.* (2005); Bambach (2008); Emmens *et al.* (2010). A high strain can be achieved in the incremental sheet forming process compared to the achievable strain in a deep drawing process. Several mechanisms that might explain the increased formability are proposed in the literature (for an overview of proposed mechanisms see Emmens and van den Boogaard (2009b)), but a definite explanation is still missing. Nowadays, the finite element method (FEM) is a powerful tool in studying and investigating metal forming processes. It provides insight details for the material during the forming process. The simplicity of the SPIF process in real-life makes it easy to create a FEM model for this process. The forming tool can be modeled by an analytical sphere, a discretized numerical blank models the workpiece and suppressing the edges of the blank models the process boundary conditions. Finally, prescribing the displacements of the numerical sphere models the displacement–controlled process.

Still, simulating the SPIF process by FEM is a major challenge. Because of the small contact area, a relatively fine mesh is used to discretize the workpiece in finite elements. Also, thousands of load increments are used to model the load history. The standard use of the well-known integration schemes (the explicit and the implicit) requires tremendous calculation times. For a small and simple academic case study, the calculation time can extend to weeks using a modern computer. The explicit time integration scheme has options that reduce the computing time significantly but the achieved results are not satisfactory. The implicit time integration scheme is accurate but it is computationally expensive. Because of accuracy, current research focuses on an efficient implementation of an implicit time integration scheme, dedicated to incremental sheet forming.

1.2 Objective and outline

The main objective of the work presented in this thesis is to simulate the incremental sheet forming process efficiently: accurate and fast. A method is proposed based on the implicit time integration scheme. Basically, the proposed method has to maintain the achieved accuracy by the implicit time integration scheme and to reduce its computational cost significantly. The proposed method is validated by simulating a demonstrative case study of SPIF. Additionally in this thesis, a fundamental study on the process mechanics of a particular type of incremental sheet forming is introduced, namely the bending under tension process.

Outline

A major part of this thesis focuses on simulating incremental sheet forming efficiently. The efficient simulation story starts in Chapter 2. A basic study on the evolution of nonlinearity in the sheet deformed by the SPIF process reveals that a localised strong nonlinearity is observed in the system of equations for the degree of freedoms that are currently located in the localised plastic deformation zone in the vicinity of the forming tool. This strong nonlinearity requires the iterative procedure of the implicit time integration scheme. The major part of the system of equations experience only a weak nonlinearity and it does not require the expensive iterative procedure. This sheds light on the fact that the standard use of the implicit time integration scheme in SPIF is inefficient with respect to the computing time. Therefore, a mixed treatment within the implicit time integration scheme is developed to treat each part as efficiently as it requires.

After the basic chapter *Aspects of SPIF modelling*, readers with different interests have more elective choices. These choices focus on different issues: the implementation of the method, the applicability of the method to other incremental forming processes and the flexibility of the method to be integrated with other numerical techniques. Being interested in the implementation of the efficient implicit method in a standard implicit scheme, Chapter 3 is the chapter to read. A super element based implementation is introduced there. Three different indicators are used to classify the super element for different treatments during the incremental procedure. These indicators are developed for the SPIF process and they are based on the current tool location, the plastic deformation in the previous increment and the change in shape.

If the applicability of the efficient implicit method for other incremental forming processes is your interest, you can read Chapter 4 after the basic chapter. The computational benefit of the efficient implicit method is measured by a speeding factor. An analytical formula is introduced to predict in advance the expected speeding factor that can be achieved by the efficient implicit method for a particular incremental forming simulation. Before implementing a single line of programing code, you can decide based on the outcome of this formula if it is (not) worth implementing the efficient implicit method.

The major interest of a developer is the flexibility of the method to be combined with other numerical techniques in order to enhance the computational performance of the method. Two numerical techniques are discussed in Chapter 5. The first technique is the static condensation. It is implemented into the efficient implicit method. A study on the performance of the enhanced method is presented. The second technique is adaptive remeshing that shows a high potential to enhance the performance of the simulation. A study on remeshing for the SPIF process is presented also in that chapter.

Two real-life incremental forming processes are simulated by the efficient implicit method in Chapter 6. The first application is to simulate the production of a pyramidal shape by the SPIF process. The second application is the simulation of multi-point incremental forming of a strip by a roll set.

Additional to the numerical part in this thesis, a fundamental study on the process mechanics of a particular SPIF process is introduced in Chapter 7. The study is carried out on a strip which is deformed by continuous bending under tension. This deformation mode has similarities with the deformation that takes place in the SPIF process. Based on a relatively simple material model, the achieved cyclic force–displacement curve of the process is explained. A numerically derived stability criterion is introduced that sheds light on the importance of bending in stabilizing the forming process. Finally, the conclusions from this research are summarized in Chapter 8.

2. Aspects of SPIF modelling

Single Point Incremental Forming (SPIF) is a challenging process to simulate. The simulation challenge is introduced in forming a blank using a small forming tool. The tool has to travel all over the blank in a lengthy forming path resulting in a slow process and tremendous simulation computing time. This chapter focuses on the numerical challenge that is summarized in simulating thousands of increments for a relatively fine FE mesh. A brief overview on the most used numerical schemes: explicit and implicit time integration is given in Section 2.1. A decision is made in favor of the implicit procedure, therefore implicit simulation of SPIF is studied in Section 2.2. In particular, the influence of localised plastic deformation on the numerical nonlinearities that are introduced in the load increments is studied. Based on that, efficient approaches are introduced in order to reduce the incremental cost of the standard Newton method. These approaches are the mixed Newton–modified Newton (NmN) approach and the coupled plastic with pseudo-linear elastic approaches: the two domain and the three domain, described in Section 2.3

2.1 SPIF modelling

Single Point Incremental Forming (SPIF) is a displacement controlled process performed on a CNC machine. A clamped blank is deformed by the movement of the tool that follows a prescribed tool path (Iseki *et al.*, 1989), a sketch of SPIF is presented in Figure 2.1. An extensive overview of the process has been given by Jeswiet *et al.* (2005); Emmens *et al.* (2010). The tool size plays a crucial role in the SPIF process for both the physical process and the numerical simulation. The small radius of the forming tool concentrates the strain at the zone of deformation in the sheet under the forming tool. The tool has to travel a lengthy forming path all over the blank to introduce the deformation resulting in a slow process in real life. The deformation in SPIF is classified as localised plastic deformation (Hirt *et al.*, 2002). According to this hypothesis, plastic deformation is localised in a small zone in the region of the forming tool surrounded by elastic deformation of the rest of the blank. The final geometry of the product is achieved by moving the local forming zone all over the blank in a lengthy toolpath. As the tool moves, a small portion of the material is plastically deformed and the material portion that just had been deformed starts to springback. This causes a simultaneous localised springback in the vicinity of the tool (Bambach *et al.*, 2009).

Numerically, SPIF requires enormous computing time regardless of the type of the solution procedures (explicit or implicit) for two reasons. First of all, modelling the sequence of small deformation increments requires thousands of numerical increments to be performed. Using too large numerical increments results in simulating a large number of penetrations

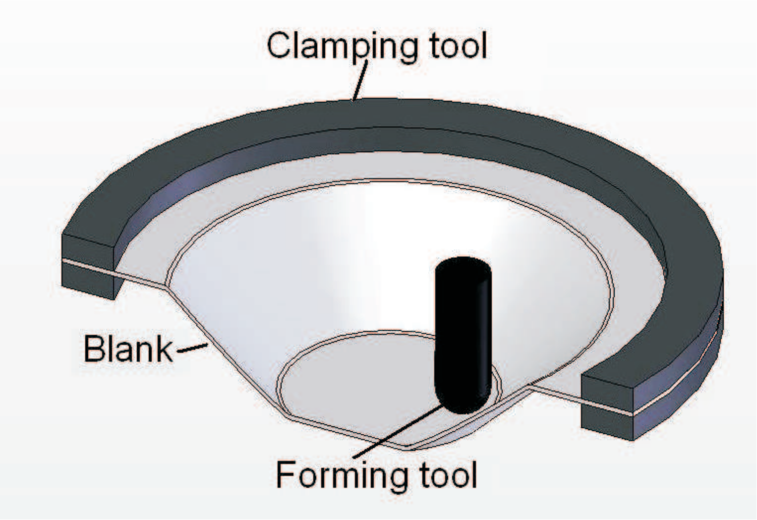

Figure 2.1: SPIF process sketch.

instead of continuous incremental forming. Secondly, the small contact area between the forming tool and the blank requires a fine FE mesh to capture the introduced deformation by the small radius of the tool. Because of the large number of numerical increments for the relatively fine FE mesh, the overall computing time for SPIF simulation is much larger than e.g. for deep drawing simulation.

2.1.1 SPIF: explicit or implicit

Both solution procedures, the explicit and the implicit time integration algorithms are available in commercial FE codes. The dynamic explicit algorithm that is based on the central difference scheme is the most used in practice. Using a diagonalized mass matrix, the explicit algorithm does not need to solve a coupled system of equations. Instead the nodal displacement and the nodal velocity are easily updated by scalar equations. No unbalance force is checked because the difference between the internal and the external force is used to determine the nodal acceleration, the velocity and then the displacement. For these reasons, the dynamic explicit method is fast and robust and these are the significant advantages of the algorithm (Belytschko *et al.*, 2007).

The major drawback of the algorithm is that it is conditionally stable. This imposes a critical, maximum, time step that can be approximated for continuum elements by the smallest time needed for a wave to cross one element. For a simulation of a material like steel, the wave speed $\sqrt{E/\rho}$ is in the order of km/s combined with element size in the order of mm that scales the critical time in order of μs (Van den Boogaard *et al.*, 2003). The time lapse in SPIF is in order of minutes to hours leading to a minimum of $10^7 - 10^8$ increments, which is prohibitively small. For this reason, explicit simulation needs more computing time than implicit method (Henrard, 2008). To overcome the small critical time step, the dynamic explicit method is enhanced by mass scaling. Increasing the mass increases artificially the material density, that decreases the wave speed and increases the critical time step. Another equivalent approach for rate independent material is time scaling e.g. to increase the forming tool velocity.

Many researchers used one of these approaches to increase the critical time step in their explicit simulation. For instance, a study on warm incremental forming shows that scaling the mass 100 times reduces the computing time of the standard explicit simulation almost by factor 8.5. The use of a larger mass scale factor results in a significant deviation of the calculated result as reported by Kim *et al.* (2008). Ambrogio *et al.* (2005b) observed a significant time reduction in explicit SPIF simulation by increasing the tool velocity artificially 2400 times. The ratio of kinetic energy to internal energy is limited to 10 % but the achieved time reduction is at the expense of accurcy and the provided results were not satisfactory. For springback analysis in deep drawing, explicit methods require the same or even more computation time as the complete forming phase (Rojek *et al.*, 1998). The implicit method can perform the springback phase in a few increments. Therefore, the forming phase is performed explicitly and often the springback is performed implicitly (Dejardin *et al.*, 2008). In conclusion, the computing time for explicit methods can be reduced significantly by mass scaling or time scaling but at the expense of accuracy.

For implicit calculations, the Newton (also called Newton–Raphson) method is the most widely used iterative method. It iterates on equilibrium of the internal and the external force using a stiffness matrix (ignoring the inertia for quasi–static processes). The major advantage of the implicit method is the unconditional stability. Because of that, the size of the increment used in an implicit method is much larger than the explicit increment size. The increment size is limited by the accuracy requirement and the robustness of the Newton procedure (Belytschko *et al.*, 2007). The implicit method is preferred for its accuracy. SPIF implicit simulations show better agreement with experiments than explicit simulations. Bambach *et al.* (2005) observed a better prediction of the achieved geometry and Ambrogio *et al.* (2005b) reported a better prediction of the sheet thinning.

The major disadvantage of the implicit scheme is the large computing time. Performing a large number of increments for a relatively fine mesh limits SPIF implicit simulation to small academic tests. Several approaches have been proposed to maintain the accuracy and to speed up the implicit simulation. For incremental forming, a multi–mesh method has been proposed. The method requires two meshes: a fine mesh for data storage and another mesh that is mainly coarse with a fine mesh part to model the deformation in the small contact area. The simulation is performed in the coarse mesh and the data is transferred between both meshes using a special operator. The computing time basically is reduced compared to the computing time of performing the simulation using the fine mesh. A recent publication of multi–mesh implementation is done by overlapping domain decomposition but only a small deformation has been introduced (Brunssen and Wohlmuth, 2009). On the thermo-mechanical simulation of a cogging process (Ramadan *et al.*, 2009), a parallel two mesh method is used. The thermal analysis is performed on a fine mesh coupled to a mechanical analysis on a coarse mesh. Significant reduction in computing time is achieved, compared to coupled analysis on the fine mesh, because the most expensive mechanical analysis is performed on the coarse mesh. The idea of decoupling is also applied on (only) mechanical problems by Sebastiani *et al.* (2007). The difference here is that the FE mesh is decoupled into an elastic part and an elastoplastic part. These separated two parts are alternately solved so that the results of one partial model provides boundary conditions for the other, a case study of small deformation is presented.

Another proposal is the use of adaptive remeshing. The basic advantage is to keep the number of degrees of freedom as low and efficient as possible that reduces the computing time compared to a fine mesh. One level of refining and coarsening is implemented by Hadoush and van den Boogaard (2008) for SPIF simulation and it is speeded up twice. Also, the use of parallel computing is reported in literatures e.g. Quigley and Monagan (2002) simulated spinning process using domain decomposition method. The previously mentioned methods for speeding implicit simulations focus on efficient modelling or the use of more computing power but not on efficient implementation of the iterative procedure. In conclusion: implicit method is accurate and expensive computing-wise but there is room to speed the procedure and maintain the accuracy. In the following sections more details are introduced to understand the Newton procedure performance in order to use it efficiently.

2.2 Implicit solution procedure

In SPIF, the tool size is much smaller than the workpiece size. The tool deforms the workpiece consequently by small increments. The small deformation increment consists of plastic deformation in the vicinity of the tool embedded in an elastic deformation of the rest of the workpiece. In implicit simulation of SPIF, the plastic deformation introduces a strong nonlinearity in the system of equations (SOE). The strong nonlinearity is a combination of material and geometrical nonlinearity. The elastically deforming part of the workpiece introduces a weak geometrical nonlinearity in the system of equations. To emphasize the strong–weak nonlinearity hypothesis in SPIF, a case study of plastic loading followed by elastic unloading of a blank (penetration test) is studied.

2.2.1 Plastic loading and elastic unloading of a blank

In this test, the strong–weak hypothesis is investigated in the simulation of tool penetration and retraction on a clamped plate. This is representative for the first and last stage of an ISF process. A plastic deformation is introduced by moving a spherical tool that is initially just in contact, 2 mm downwards. Then the blank is relaxed by moving the tool away. The deformation and the relaxation are performed in 20 increments and 5 increments (a load increment of 0.1 mm is used), respectively. The FE mesh and position of the tool are shown in Figure 2.2.

The numerical blank of $100 \times 100 \times 1.2\,\text{mm}^3$ is discretized with 6400 triangular shell elements. The element type is the discrete Kirchhoff triangle DKT for bending (Batoz *et al.*, 1980), combined with a linear membrane element. The element has 6 DOFs per node, 3 translational DOFS (U_x, U_y, U_z) and 3 rotational DOFS $(\theta_x, \theta_y, \theta_z)$. It has 3 integration points in plane and 7 in thickness direction (in total 21). The tool is modelled by a 20 mm diameter analytical sphere. The material model is representative of mild steel and it is kept as simple as possible. The isotropic yield behavior of the material is modelled with the von

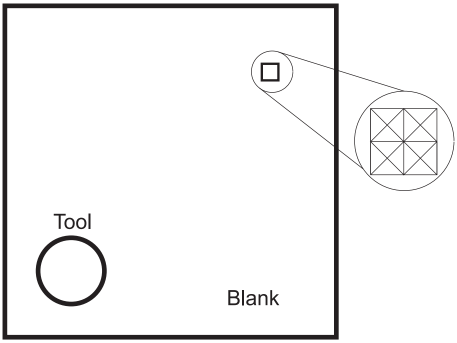

Figure 2.2: A sketch of the penetration test, a sample of FE mesh and the tool position.

Mises criterion. The work hardening is governed by the power law:

$$\sigma = 500(\varepsilon + 0.00243)^{0.2} \tag{2.1}$$

Where σ and ε are the flow stress and the equivalent plastic strain, respectively. The material has a Young's modulus of 200 GPa and Poisson's ratio of 0.3. For a realistic calculation, it is acknowledged that a better material model is required, that includes e.g. the anisotropic behavior of the sheet. The calculated vertical force on the tool is plotted in Figure 2.3. In the loading stage, the plate is deformed plastically near the tool and a nonlinear prediction of the force is observed. In the unloading stage, the plate shows elastic springback.

The simulation is implicitly performed using the Newton iterative procedure implemented in the in–house FE code DiekA. A mechanical unbalance ratio of 0.001 is used for checking the convergence. The number of iterations required per increment during the simulation is plotted in Figure 2.4. For the loading stage, most of the increments require 3 iterations per increment (on average 2.8 iterations/increment). During unloading, the first unloading increment, increment number 21, requires 6 iterations and 2 line searches because of the sharp transition of loading–unloading. All the unloading increments require more than 1 iteration to converge hence a geometrical nonlinearity is involved. Within the increment before the last, the tool–blank contact is lost and that explains the kink in the unloading path. The last unloading increment requires 2 iterations because of the use of relative unbalance criterion. Actually, the unbalance force is very small. The incremental cost is the multiplication of iteration cost by the number of iterations consumed in the increment. The iteration cost, on average, is 2.45 s. As expected, increment 21 has the largest CPU time of 15 s since it requires the largest number of iterations. The total CPU time is 179.3 s. This simulation is performed on a single core of Sun Fire $X4450$ server with Intel Xeon $X5365$, this machine will be used for all simulations presented in this thesis unless another machine is mentioned.

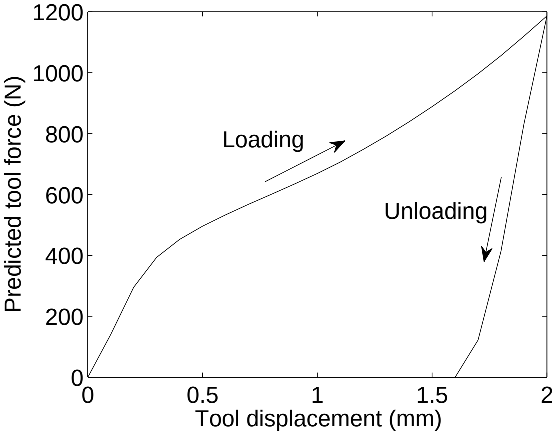

Figure 2.3: The predicted force displacement curve of the penetration test.

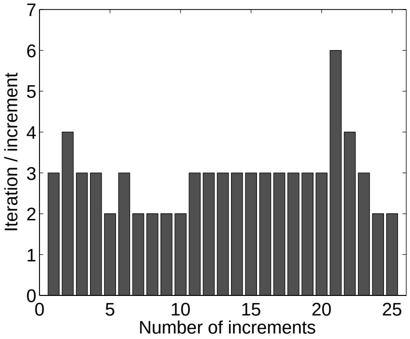

Figure 2.4: Number of iterations for the penetration test.

2.2.2 Strong–weak nonlinearity

The test introduced in the previous section is used to emphasize the strong–weak nonlinearity. Incrementally, a small region of the blank is plastically deformed and it is located in the vicinity of the forming tool. The rest of blank is elastically deforming. The achieved equivalent plastic strain at the end of the loading stage is shown in Figure 2.5. The presented result of the equivalent plastic strain is related to the upper integration point in thickness and the same distribution is observed for the rest of the integration points through thickness. The maximum achieved equivalent plastic strain is 0.169. Near the close edges a

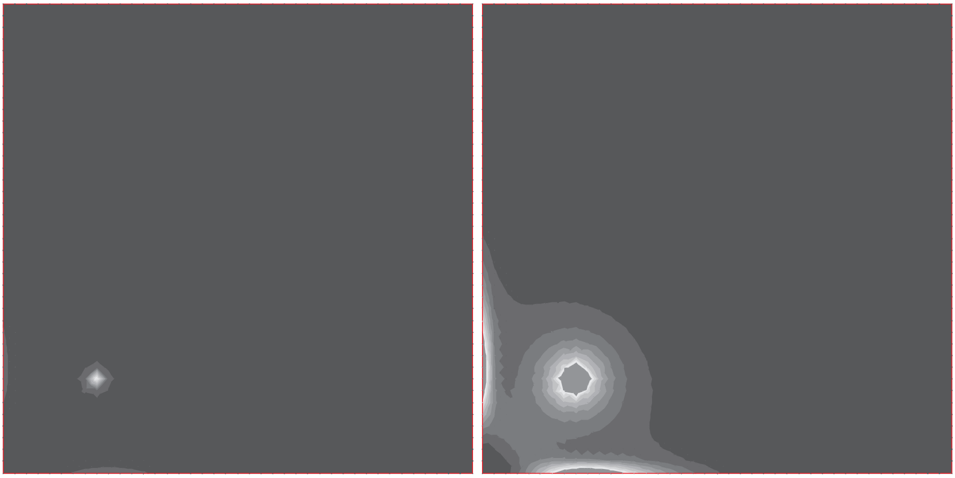

Figure 2.5: The achieved upper equivalent plastic strain, left the scale is $0 - 0.17$. The scale is reduced to $0.001 - 0.02$ in the right figure.

relatively small plastic strain is observed. The plastic deformation is a form of material nonlinearity. The localised plastic deformation near the tool indicates that this part of the material experiences a higher level of stress compared to the rest. The nonlinearity due to elastic–plastic transition is much larger than the nonlinearity due to change of shape. The vertical displacement at the end of the loading stage is plotted in Figure 2.6. Clearly, a large displacement gradient is noticed near the tool resulting in a large rotation that is the main geometrical nonlinearity. The geometrical nonlinearity effect in the blank is strong in the vicinity of the tool and it is relatively weak away from the tool.

As a consequence of the localised plastic deformation near the tool, a combination of material and geometrical nonlinearities forms a strong nonlinearity. The rest of the blank experiences a weak geometrical nonlinearity and it will be referred to as weak nonlinearity. To study the strong–weak nonlinearity during the iterative procedure, the residual force (unbalance between the external force and internal force) of 2 nodes are recorded for the entire simulation. One of these nodes is located under the tool (strong) while the other one is located at the center of the blank (weak). For the same node, the residual force in vertical direction has the most significant residual contribution compared to the other DOFs, for that the residual force in vertical direction is plotted in Figure 2.7. Noticing the logarithmic scale, it becomes clear that the DOF in the vicinity of the tool has a large residual value and it is reduced significantly with the iterative procedure. The residual has to be reduced to a certain tolerance. The residual of the central DOF in the weak nonlinearity region is negligible compared to the residual value in the strong nonlinearity region. This holds for all increments.

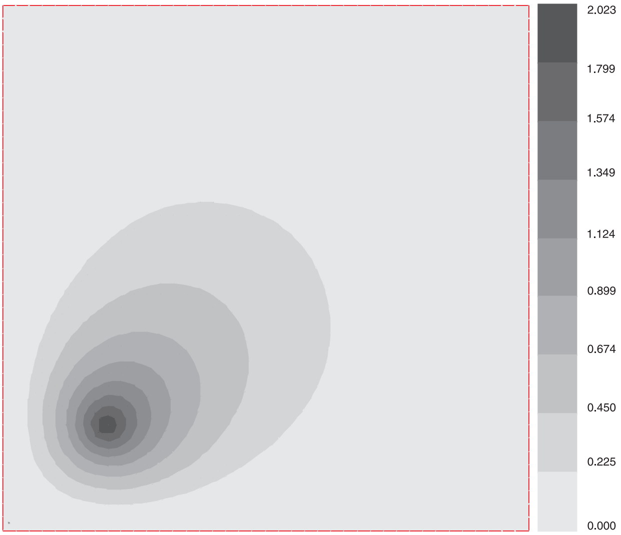

Figure 2.6: The vertical displacement at the end of the loading stage of the penetration test.

2.3 Efficient implicit simulation of localised deformation

Based on the classification of strong and weak nonlinearities in the previous section, it is clear that the strongly nonlinear part of the system of equations requires fully nonlinear iterative treatment. It is an expensive treatment. The rest of the system of equations represents a large elastic part, which does not need such expensive treatment but it has to be created to solve the system of equations. For the sake of understanding, the implicit scheme is summarized briefly.

The Newton–Raphson method updates an incremental displacement vector d with an iterative displacement vector Δd, using the tangent of the nonlinear system of equations $K(d)$ by solving

$$R(d) + K(d)\Delta d = 0 \tag{2.2}$$

where the residual $R(d)$ defines the difference between the internal forces and the external forces

$$R(d) = f_{\text{int}}(d) - f_{\text{ext}}(d) \tag{2.3}$$

The Jacobian system matrix $K(d)$ or in engineering terms the effective tangent stiffness matrix (stiffness matrix), is equal to

$$K(d) = \frac{\partial R}{\partial d} = \frac{\partial f_{\text{int}}}{\partial d} - \frac{\partial f_{\text{ext}}}{\partial d} = K_{\text{int}} - K_{\text{ext}} \tag{2.4}$$

where K_{int} and K_{ext} are the tangent stiffness matrix and the load stiffness matrix, respectively.

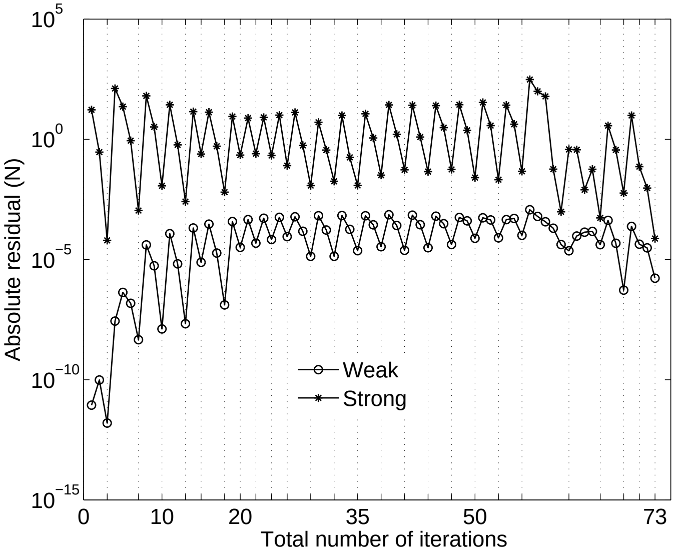

Figure 2.7: The residual evolution, the vertical dashed line indicates the end of an increment, the marker indicates the iterative residual value.

The linearized model is solved for the iterative change of the nodal displacements Δd

$$\Delta d = -K^{-1}R \tag{2.5}$$

the iterative change of the nodal displacements is added to the total incremental nodal displacements

$$d^{j+1} = d^{j} + \Delta d \tag{2.6}$$

where j is the iteration number. If convergence is not achieved, the linearized model is recalculated and solved for a new Δd. Here, the residual is checked for convergence by the mechanical unbalance ratio criterion. The mechanical unbalance ratio ψ is the ratio of the l_2 norm of the residual to the l_2 norm of the internal force

$$\psi = \frac{\|R\|}{\|R_{int}\|} \tag{2.7}$$

The Newton iteration cost can be split into three parts (Van den Boogaard *et al.*, 2003). The first part creates the linearized model (2.2) this includes the creation of the tangent stiffness matrix and the internal force vector (BUILD). Secondly, solving the system of equations (SOLVE) for the iterative displacement (2.5). The last part is to update the stresses based on the actual displacement (UPDATE). This means that a large part of the computing power is used inefficiently for updating the large elastic part. Using a relatively less expensive iterative procedure like the modified Newton to create the entire system of equation does not reduce the overall computing time even though it reduces the cost of the BUILD phase on the iteration level. Based on experience, the modified Newton method requires a large number of iterations per increment and smaller increment sizes to converge compared to the full Newton approach.

For such localised numerical nonlinearities in system of equations, it becomes necessary to have different treatments that are accurate and computationally efficient for different parts of the FE mesh. Similar approaches are reported in the literature for mixed treatment in computational mechanics e.g. the subcycling in explicit methods to overcome the problem of very small or very stiff elements Belytschko *et al.* (1979). Another approach is the implicit–explicit method, where part of the system Jacobian matrix is treated implicitly and part explicitly Hughes and Liu (1978). In the following sections, the internal force vector and the tangent stiffness matrix for the localised plastic deformation part are updated for every iteration using the full Newton method. For the elastically deforming part of the FE mesh, the internal force vector and the tangent stiffness matrix are treated either by modified Newton method or pseudo-linear approach. The entire system of equations is solved for each iteration, but the domains are treated differently. The purpose of such treatment in the localised deformation implicit simulation is to reduce the overall CPU time. The implementation and testing is done in the in–house FE code DiekA.

2.3.1 Mixed Newton–Modified Newton

As it is observed in SPIF, the system of equations is assembed of two types of DOFs with respect to the nonlinearities. The first type experiences a strong nonlinearity and the second type has a weak nonlinearity. It is recommended to have fully nonlinear Newton treatment for the strong type nonlinear DOFs because of its quadratic convergence. In this treatment, K and f_{int} are updated every iteration including the geometrical and the material nonlinearities. Contact, or changing the boundary conditions, introduces a nonlinearity in the system of equations even for the linear elastic system. The DOFs near the tool have high chances to make contact or to lose contact with the forming tool. Therefore, iterative treatment is necessary for these DOFs in order to predict the contact. The weak nonlinearity in the second type of DOFs is treated by the modified Newton method. In this treatment, the f_{int} is updated fully nonlinear and this is similar to the Newton approach.

The difference between the Newton method and the modified Newton method is the treatment of the effective tangent stiffness matrix. In the modified Newton method, K is not updated iteratively, instead a previously calculated K is reused. This previously calculated stiffness matrix might be calculated at the start of the increment or several increments before (Zienkiewicz and Taylor, 2005). Here, the stiffness matrix is calculated at the beginning of each increment including the geometrical and material nonlinearities. Within only one increment, the stiffness matrix is reused and it is updated nonlinearly at the beginning of the next increment. It is worth mentioning that the residual and the stiffness matrix of the FE mesh that is treated by the modified Newton method has no external contribution. Iteratively, the global effective tangent K_{Glob} is assembled of the iteratively updated stiffness matrix and the incrementally updated stiffness matrix

$$K^{\mathrm{Glob}} = \sum_{e=1}^{ITE} K_{\mathrm{e}}^{\mathrm{Iter}} + \sum_{e=1}^{INE} K_{\mathrm{e}}^{\mathrm{Incr}} \tag{2.8}$$

where the superscript abbreviation Glob, Iter and Incr are global, iterative and incremental, respectively. Actually, it is an assembly operation not a simple summation operation and it is used for convenience. The assembly operation is performed over the number of iterative elements ITE and the number of incremental elements INE for the iterative and incremental stiffness matrix, respectively. The global residual force vector is assembled

$$R^{\mathrm{Glob}} = \sum_{e=1}^{ITE+INE} R_{\mathrm{e}} \tag{2.9}$$

the residual force vector for all elements in FE mesh is iteratively updated. Now, the entire system of equations is solved for Δd (2.5) and the total increment is updated (2.6). The new internal force vector is found and the convergence is checked (2.7).

This mixed Newton–modified Newton (NmN) approach is applied to the penetration test. The FE mesh is classified into a strong nonlinear part, in the vicinity of the tool, that is colored in gray in Figure 2.8 and the weak nonlinear part (white), presenting the rest of FE mesh. The predicted tool force by the NmN approach is, almost, equal to the prediction achieved by the full Newton approach, Figure 2.9. The maximum error is less than 0.02 N (0.0025 %) and it has been observed in the unloading stage. This excellent agreement is achieved by classifying the right elements in the strong nonlinearity group, applying the full Newton treatment. The residual history of R_{z} in the strong nonlinearity region is preserved for amplitude and pattern as shown in Figure 2.10. The same number of iterations required by the Newton simulation (Figure 2.4), and line search, is consumed by the mixed Newton–modified Newton approach and in the same order.

The total CPU time of the mixed Newton–modified Newton approach is 157 s that is 22.3 s less than the full Newton approach (179.3 s). In the Newton approach the costs of the main parts are 1.18 s (48.6%), 0.28 s (11.5%) and 0.97 s (39.9%) for BUILD, SOLVE and UPDATE, respectively. The reduction in the overall CPU time is achieved by reducing each increment cost as plotted in Figure 2.11. In this case study, the NmN approach applies fully iterative treatment for 36% of elements (gray area in Figure 2.8) and 64% of the elements are treated by the modified Newton method. The cost of the first iteration in the NmN approach is the same as the Newton approach and that is independent of split ratio of elements into Newton or modified Newton within NmN. After the first iteration, the BUILD CPU cost, in NmN, is reduced because the stiffness matrix of 64% of the elements is not calulated again. The BUILD CPU cost becomes the cost of calculting the stiffness of 36% of elements and the force vector of all elements and it is 0.73 s. The cost of SOLVE is similar for all iterations because NmN has no interaction with the solver or the system of equations size. Also, the UPDATE cost is similar because the modified Newton method is equal to the Newton method with respect to UPDATE. The reduction in NmN's incremental cost is a result of reducing BUILD cost only. The total reduction of the NmN incremental cost is equal to the reduction in BUILD cost times the number of iterations except the first one. A

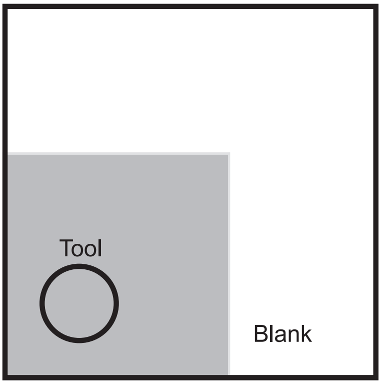

Figure 2.8: Sketch of FE mesh classification into Newton treatment (gray) and modified Newton or pseudo-linear treatment (white).

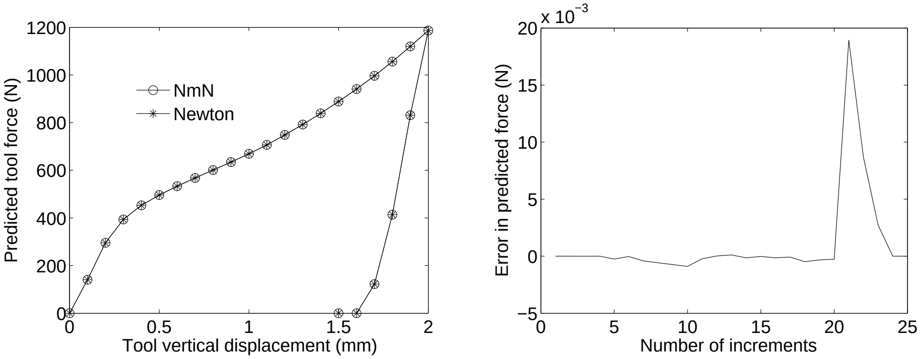

Figure 2.9: A comparison of the predicted force displacement curve by mixed Newton–modified Newton approach and Newton approach (left), error evolution during the simulation (right).

large number of iterations per increment increases the reduction in incremental cost. The maximum reduction in NmN incremental cost is 2.3 s for increment number 21 which uses 6 iterations. To this end, a reduction is observed in the standard Newton incremental cost by applying the mixed Newton–modified Newton approach and the predicted results have excellent agreement with the results achieved by the full Newton method.

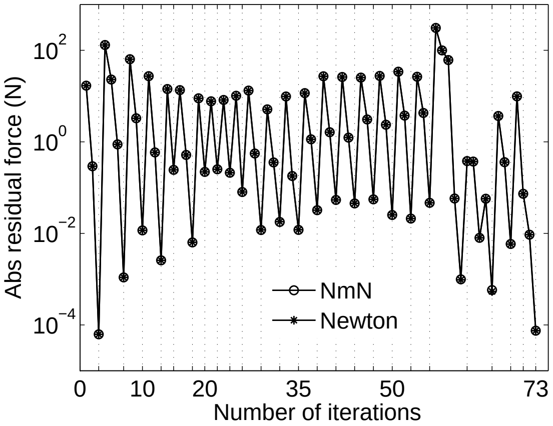

Figure 2.10: The strong nonlinearity evolution in mixed Newton–modified Newton and Newton approaches.

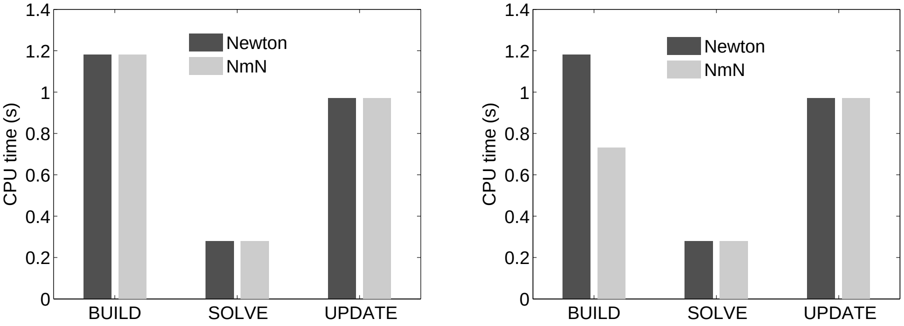

Figure 2.11: A comparison of the iterations cost between the mixed Newton–modified Newton approach and the Newton approach. The first iteration of each increment (left) and the other iterations (right). The results are presented for increment number 15.

2.3.2 Two domain approach

In this approach, the FE mesh is split into two parts as in the mixed Newton–modified Newton approach. The first part contains the strong nonlinearity in the vicinity of the tool (the gray area in Figure 2.8). It is an iterative part that is nonlinearly updated and predicts the plastic deformation iteratively. The second part models the elastically deforming part of the blank and it is treated pseudo-linearly. It models linear elastic deformation within the number of increments. The nonlinearity is updated at the beginning of the increment or group of increments. Here, the pseudo-linear treatment is applied incrementally. At the beginning of the increment, the stiffness matrix and the internal force vector is calculated fully nonlinear that includes the geometrical and the material nonlinearities of the previous increments. This applies for the entire FE mesh (the plastic and the pseudo-linear elastic part). The linearized model is assembled, there is no difference in treatment between the plastic and the elastic parts, to this point. The system is solved at once for Δd.

In the strong nonlinearity zone, the new stress state is nonlinearly updated. This is an expensive procedure because an iterative procedure is used to find the balance between the elastic and plastic strain, it is often referred to by return mapping algorithm. This procedure is performed on the integration point level. The new internal force vector is determined and the contribution in the residual vector is created. The weak nonlinearity zone (elastically deforming) is treated with a less expensive approach. The stresses are assumed to be linearly and elastically related to the strains. As a consequence, the internal force vector is updated linearly by the multiplication of the stiffness matrix and the incremental displacements as

$$f_{\text{int}}^{j} = f_{\text{int}}^{0} + K_{\text{int}}^{0} d^{j} \tag{2.10}$$

the residual contribution of the linear elastic has no external force contribution. The global residual is assembled

$$R^{\text{Glob}} = \sum_{e=1}^{ITE} R_{\text{e}}^{\text{Iter}} + \sum_{e=1}^{INE} R_{\text{e}}^{\text{Incr}} \tag{2.11}$$

The convergence is checked and often more iterations are required. In the following iteration, K and f_{int} of the plastic part are nonlinearly updated. The K of the elastically deforming FE is not updated and kept constant as it is treated in the mixed Newton–modified Newton approach while the f_{int} is linearly updated as in (2.10) instead of being updated nonlinearly. The K_{Glob} is assembled of the iteratively updated part and the incrementally constant part as in (2.8). The residual force vector is assembled of the iteratively updated part and the linearly updated elastic part as in (2.11). The linearized model is created and solved and so on.

The performance of the two domain approach is tested using the penetration test. The predicted tool force by the two domain approach has a very good agreement with the prediction achieved by the Newton approach as shown in Figure 2.12. The maximum error in the force prediction is observed during the unloading stage. A maximum error of 0.9% (less than 2 N) at the third unloading increment is found, which is within the acceptable limit. During the loading stage, the same number of iterations per increment is used by the two domain approach as in the Newton approach. The R_{z} convergence behaviour of the strong nonlinearity by the two domain approach coincides with the prediction by the Newton approach, Figure 2.13. The two domain has a similar behaviour as the Newton approach during the unloading stage except for a slight difference for the first unloading increment. Both approaches perform similarly for the first 3 iterations of the first unloading increment. At the fourth iteration, The two domain has a global unbalance of 0.00098 while the Newton has a global unbalance of 0.00102, a convergence tolerance of 0.001 is used in both simulations. The Newton approach requires one more iteration to converge. At

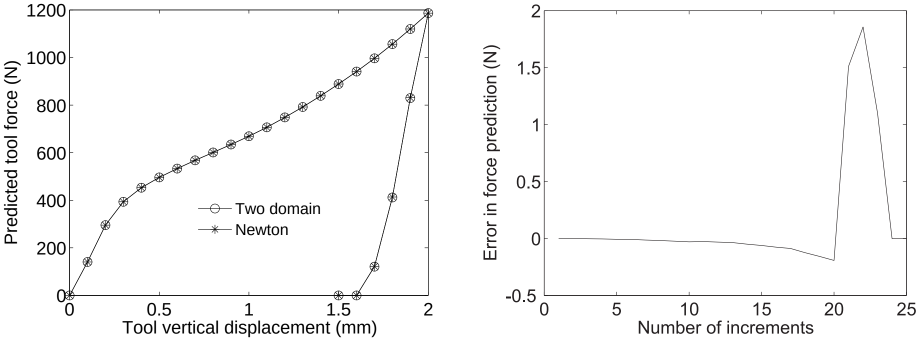

Figure 2.12: A comparison of the predicted force displacement curve by the two domain approach and the Newton approach (left), error evolution during the simulation (right).

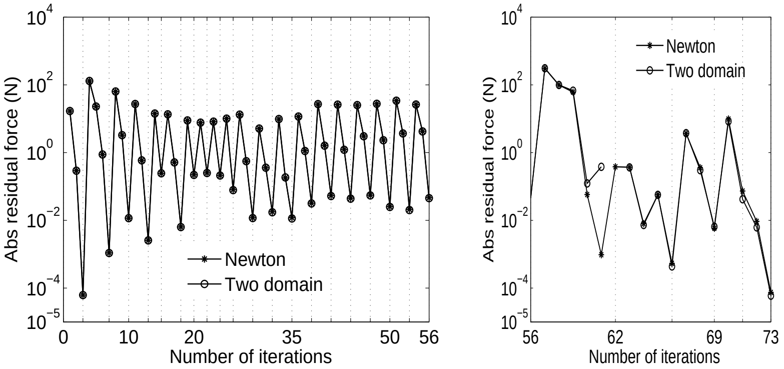

Figure 2.13: The strong nonlinearity evolution in two domain and Newton approaches for the loading stage (left) and the unloading stage (right). The vertical grid indicates the end of the increment in the Newton approach.

convergence, both approaches have not reached the contact convergence. Another iteration is required to achieve contact convergence. In this iteration, an increased value of R_z residual is observed but the global convergence is already achieved.

The overall CPU time of the two domain approach is 121.7 s, that is 57.6 s less than the CPU time of the Newton approach. The incremental cost is significantly less than the incremental cost of the Newton approach as shown in Figure 2.14. The reduction in incremental cost is a result of reducing each iteration cost. Considering increment number 15, the cost of UPDATE is reduced from 0.97s to 0.54s (-44.3%) by two domain, for each

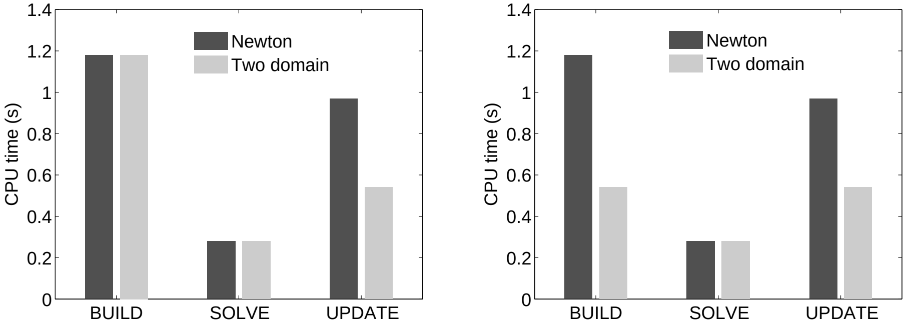

Figure 2.14: A comparison of the iterations cost between two domain approach and Newton approach. The first iteration of each increment (left) and the other iterations (right). The results are presented for increment number 15.

iteration. The two domain cost of BUILD is equal to the Newton cost of BUILD for the first iteration only while a reduction of −54.2% for BUILD is achieved (from 1.18s to 0.54s) by two domain for the later iterations.

The advantage of the two domain approach over the mixed Newton–modified Newton approach is related to the stress update procedure in the large elastically deforming FE part. In the mixed Newton–modified Newton approach, the stress is nonlinearly updated using return mapping algorithm (expensive procedure). For the two domain approach, it is assumed to be linearly and elastically related to the strain therefore is not updated within the increment. This less expensive treatment reduces the UPDATE iteration cost significantly even for the first iteration of each increment. After the convergence of the increment, a fully nonlinear update of the stress state is performed based on the displacement increment. This nonlinear evaluation updates the small material and the geometrical nonlinearity. The material update may introduce a plastic deformation in the proposed elastically deforming FE mesh that is not checked for equilibrium. Therefore, the size of the plastic region has to be selected carefully to accurately model the introduced deformation. The cost of updating the stress state of the pseudo-linear domain is part of the two domain incremental cost and it is performed once per increment.

2.3.3 Three domain approach

The new part in this approach is the split of the pseudo-linear treatment of the weak nonlinearity zone (elastically deforming) into two parts. The first part is a pseudo-linear treatment within one increment and the nonlinearities is considered at the start of the increment only. The second part is similar to the first part except that the linearity is assumed for a group of increments instead of one increment. Now, the FE mesh of the entire model is split into 3 domains as shown in Figure 2.15. The first domain is treated iteratively fully nonlinear,

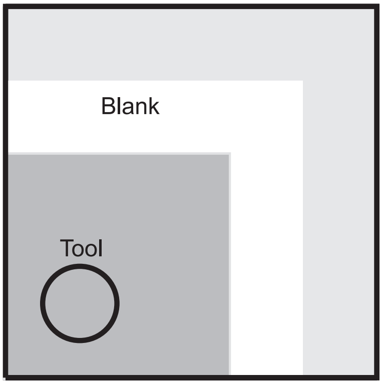

Figure 2.15: Sketch of FE mesh classification into Newton, iterative, treatment (gray) and incremental pseudo-linear treatment (white) and multi-incremental pseudo-linear treatment (light gray).

the gray colored part. The second domain applies an incremental pseudo-linear treatment (white part). The multi-incremental pseudo-linear domain (light gray) models the last part.

The penetration test is performed by the three domain approach. The FE mesh is split into 36% iteratively, 28% incrementally and 36% multi-incrementally updated treatment. This reduces the overall CPU time to 110 s that is 69.3 s less than the Newton approach and 11.7 s less than the two domain approach. The lower computing time required by the three domain compared to two domain is achieved because of updating 43.75% of the elastically deforming part by multi-incremental and 56.25% by the incremental pseudo-linear treatment. In the multi-incremental domain, the stiffness matrix and internal force vector are calculated only once for the entire simulation. The internal force vector is linearly updated as in (2.10) by the multiplication of the stiffness matrix and the corresponding total incremental displacements. The evolution of iteration of each increment is similar to the Newton approach. A very good agreement is achieved in the predicted force–displacement curve as shown in Figure 2.16 with error less than 0.25% (2 N). To conclude, the two domain and three domain approaches speed up the standard Newton method by a factor of 1.47 and 1.63, respectively. An important aspect influencing the speeding factor is the ratio of the elements that are iteratively, incrementally or multi-incrementally treated.

2.4 Summary and conclusions

In this chapter, the challenge of simulating SPIF process is presented. The challenge, simply, is using a small size forming tool that introduces plastic deformation locally. This requires performing thousands of load increments on a relatively fine FE mesh resulting in enormous computing time regardless of the used numerical procedures: explicit or implicit. The explicit scheme CPU time can be reduced significantly by the use of mass scaling or time

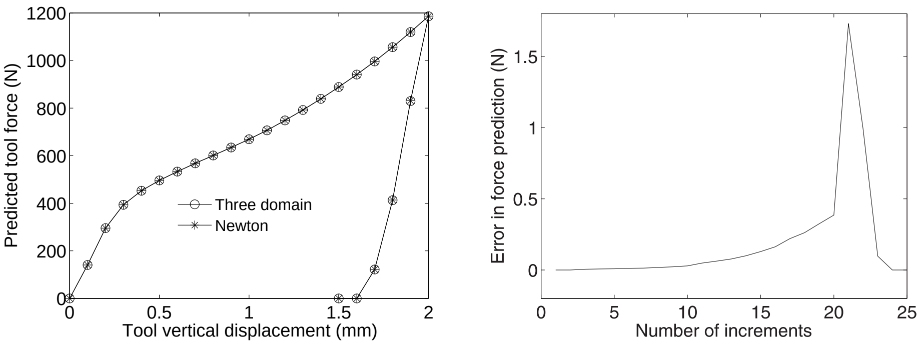

Figure 2.16: A comparison of the predicted force displacement curve by the three domain approach and Newton approach (left), error evolution during the simulation (right).

scaling. This reduction in the overall CPU time of explicit simulation is at the expense of the achieved accuracy. It is observed that the implicit SPIF simulation is more accurate than the explicit in predicting the final geometry and the sheet thinning. The implicit incremental CPU time is expensive because of the iterative aspect that by itself is expensive. Several approaches are proposed in order to reduce the SPIF implicit CPU time. Mainly, these approaches focus on efficient modeling or the use of more computing power.

Because of the localised plastic deformation, part of the FE mesh that is in the vicinity of the tool experiences a strong nonlinearity. The strong nonlinearity is a combination of the material and geometrical nonlinearities. The rest of the FE mesh that models the elastically deforming part of the blank experiences only a weak nonlinearity. It is required to use the standard Newton method because of the strong nonlinearities in the system of equations but it is inefficiently used for the large elastically deforming part. Therefore, it becomes necessary to have different treatments that are accurate and computationally efficient for different part of the FE mesh. The fully nonlinear Newton treatment is used for the localised plastic deformation. The rest of the FE mesh that is elastically deforming is treated either by the modified Newton method or the pseudo-linear approach. The purpose of such treatment in implicit simulation of SPIF is to reduce the overall CPU time. The implementation and testing is done in the in–house FE code DiekA.

A case study of localised deformation of a blank using small tool is studied. The overall computing time for all used approaches is summarized in Table 2.1. Different speeding factors are achieved based on the treatment of the elastically deforming part. The mixed Newton–modified Newton speeds up the Newton simulation by 1.14. This is achieved by the use of the modified Newton approach. The two domain approach applies a pseudo-linear treatment that has a linear treatment within the increment while the nonlinear treatment is applied only at the beginning of each increment for the material and geometrical nonlinearities. This results in speeding the Newton simulation by 1.47. The best performance of speeding the Newton simulation is achieved by the three domain approach (1.63) because

Table 2.1: The overall computing time for different approaches. The speeding factor is defined as the ratio of Newton CPU time, reference, to a particular approach CPU time.

	Newton	NmN	two domain	three domain
CPU time (s)	179.3	157	121.7	110
Speeding Factor	1.00	1.14	1.47	1.63

the elastically deforming part is split into two parts. Both parts are treated pseudo-linearly, one part is incrementally and the other is for all increments that is even less expensive. In all proposed approaches, the error in predicting the tool force is less than 1%. The most accurate approach is the mixed Newton–modified Newton approach. It is clear now that the SPIF implicit simulation cost can be reduced by applying efficiently treated zones as required.

But, several important issues are still open to investigation. First of all, the definition of partitions that are treated differently is presented for a relatively simple case, fixed in–plane tool position, while the tool path in a SPIF process is more complex. The ratio of the partitions has an impact on accelerating the SPIF implicit simulation and it is clearly defined based on experience. Therefore, automated features have to be introduced to notify the distribution (location) and the optimized size of these partitions in order to simulate the localised plastic deformation efficiently in the simulation of a SPIF process.

3. Adaptive domain classification

In the previous chapter, an efficient implementation of the implicit solution procedure for localized deformation is introduced. A fixed partitioning of the FE mesh is used for a simple test in which a blank is plastically deformed by a small tool followed by springback. The special aspect of incremental sheet forming requires that the small sized forming tool travels all over the blank in order to introduce the global deformation. For this reason, this chapter focuses on introducing a procedure that controls the partitioning of the FE mesh. The main tasks of this procedure are firstly to define the size of each partition and secondly to define the location and the distribution of these partitions. The major components of this procedure are super elements and indicators. The main task of a super element is to organize and manage the data that are used in the implicit procedure, in Section 3.1. Super elements are generically classified regarding the update frequency (iterative, incremental or multi-incremental). This is performed by introducing indicators that can define the super elements classification prior to load increment. The indicators are based on the current tool location, the plastic deformation during the previous load increment and the shape change during the previous load increment. The indicators are discussed in Section 3.2. The performance of the efficient implementation of the implicit procedure combined with the introduced indicators is investigated by two case studies of incremental forming processes. The first case is to simulate one loop of SPIF and the second case is to simulate continuous bending under tension in Section 3.3.

3.1 Super element (substructuring)

The basic idea of grouping a particular FE mesh into several super elements, substructures or domains is often used in the finite element method for different purposes. It is adopted early in structural analysis where a complete structure is partitioned into a number of substructures that can be treated as complex structural elements. Internal displacements of the substructure are condensed to the external (boundary) displacements then the displacement method, for instance, is applied to the partitioned structure. Each substructure can then be analyzed separately under known substructure–boundary displacements or forces (Przemieniecki, 1968). Another use of super elements is to present complex element behavior e.g. a general beam finite element with deformable cross section is introduced by Živković *et al.* (2001). This special element is formulated as a super element that consists of isoparametric subelements (3D, shell and beam). In this thesis a super element is used in order to organize the elements tangent stiffness matrix, internal force vector and degrees of freedom. Also, it facilitates the management of updating these data for each iteration, increment or number

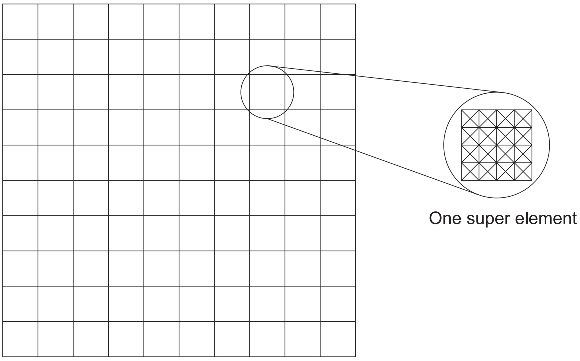

Figure 3.1: FE mesh of 6400 triangular shell elements partitioned in 100 super elements.

of increments for an efficient implementation of the implicit time integration procedure. A super element can be one element or a group of elements. A particular FE model is represented by a group of super elements instead of elements. An example is shown in Figure 3.1.

The super element tangent stiffness matrix K^{Super} is assembled of the related element stiffness matrices as

$$K^{\text{Super}} = \sum_{e=1}^{SE} K_{\text{e}} \tag{3.1}$$

where SE is the number of elements grouped in the super element. The global tangent stiffness matrix K^{Glob} is assembled of super element tangent stiffness matrices

$$K^{\text{Glob}} = \sum_{s=1}^{SS} K_{\text{s}}^{\text{Super}} \tag{3.2}$$

where SS is the number of super elements in the FE mesh. The super element internal force vector $f_{\text{int}}^{\text{Super}}$, and the global internal force vector $f_{\text{int}}^{\text{Glob}}$are assembled similarly

$$f_{\text{int}}^{\text{Super}} = \sum_{e=1}^{SE} f_{\text{int,e}} \tag{3.3}$$

$$f_{\text{int}}^{\text{Glob}} = \sum_{s=1}^{SS} f_{\text{int,s}}^{\text{Super}} \tag{3.4}$$

The super element contains also the related incremental displacement d.

Up to this point, the system of equations is assembled of the super elements contributions and regardless of the frequency of updating these contributions (tangent stiffness matrix and internal force vector) and the distribution of the super element based on the update frequency.

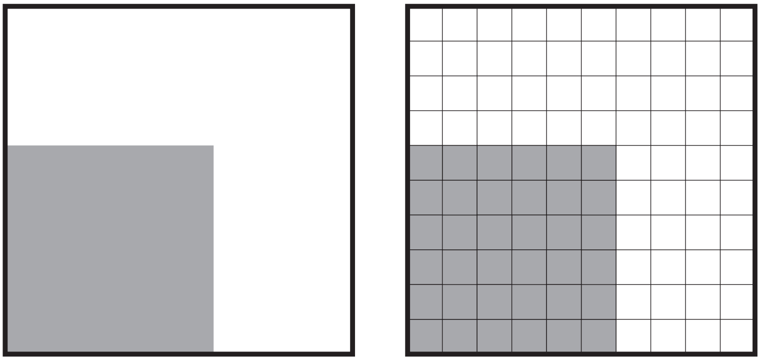

Figure 3.2: FE mesh partitioned into two domains of different update treatment (left). The same two domains are modeled by several super elements (right) that is 36 and 64 for the gray domain and the white domain, respectively.

This re–arranges the layout of the implicit time integration procedure and facilitates the implementation of the efficient implicit procedure based on super elements. By the use of super elements, the FE mesh partitioning presented in Figure 2.8 is modeled by several super elements per domain, as shown in Figure 3.2. A super element can belong only to one partition since each partition has a different frequency of updating the super element contributions. Also, the FE mesh is grouped into non–overlapped super elements. It is important to mention that the size of the system of equations is similar to its original size and the entire system of equations is solved at once.

3.1.1 Implementation of the efficient implicit approach

The efficient implicit approach, introduced in the previous chapter, is implemented by the use of super elements. The approach splits the FE mesh into iteratively, incrementally and multi-incrementally updated domains and it accelerates the simulation compared to the standard Newton simulation that updates the entire FE mesh every iteration. The iterative treatment updates geometrical and material nonlinearity for the tangent stiffness matrix and the internal force vector for each iteration. Updating all super elements in a FE mesh iteratively results in achieving exactly the same results achieved by the standard Newton method to machine precision and introduces no reduction in the simulation CPU time. The iteratively updated treatment is recommended for the plastically deforming domain of the FE mesh.

The incrementally updated treatment of a super element applies pseudo-linear behavior within the load increment. The tangent stiffness matrix and the internal force vector are calculated at the beginning of each increment and that includes the geometrical and the material nonlinearity. The super element tangent stiffness matrix K^{Super} is assembled once and it is used for all iterations within the increment, i.e. it is considered constant within

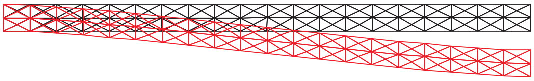

Figure 3.3: The initial FE mesh of a strip (top) and the FE mesh at the end of the deformation (bottom).

the increment. The super element internal force vector $f_{\text{int}}^{\text{Super}}$ is calculated from the actual stresses at the beginning of the increment. Then, it is updated linearly within the increment by the multiplication of K^{Super} with the super element's incremental displacement vector as presented in (2.10). At the beginning of next load increment, an update of the geometrical and material nonlinearity is performed for K^{Super} and $f_{\text{int}}^{\text{Super}}$. This update is based on the converged load increment and the incremental displacements. All elements contained within the super element are updated. This treatment is recommended for the elastically deforming part of the FE mesh that experiences a weak nonlinearity because of geometrical effects (change of shape). It is less expensive compared to the iterative treatment. The multi-incremental update procedure is similar to the incremental update treatment except it is performed over a number of increments instead of over one increment.

Validation

A simple test is proposed to investigate the influence of the linearization in an incrementally treated super element with respect to geometrical nonlinearity. For this purpose, an initially flat strip of $50 \times 5 \times 1.2\,\text{mm}^3$ with elastic material is modelled with 160 shell elements as plotted in Figure 3.3. The edge on the left of the strip is completely clamped. The edge on the right is moved 5 mm downwards while the rotations and in–plane translation degrees of freedom are suppressed. Four increment sizes are considered 0.01, 0.1, 0.2 and 0.5 mm.

The FE mesh of the strip is grouped in one super element. Two simulations are carried out per increment size. In the first simulation, the geometrical nonlinearity is treated iteratively like in the standard Newton implicit method (Iterative). In the second simulation the super element is updated incrementally (Increment). The geometrical nonlinearity is neglected within the increment and updated nonlinearly at the beginning of each increment as explained before.

To compare the results, the in-plane force (longitudinal) at the right edge for a displacement increment of 0.2 mm is plotted in Figure 3.4. The achieved force by the incrementally updated approach has a very good agreement with the standard Newton (iterative) approach with a maximum error of 1.7%. The achieved force by the incrementally updated approach for the other increment sizes has the same pattern, the error is plotted in Figure 3.5. As expected, the largest increment size of 0.5 mm gives the largest error of 4%. In SPIF simulations the displacement per increment will usually be much smaller. Under these conditions, the incrementally linearized approach can be considered valid. Similarly, the multi-incremental linearized approach is validated for smaller increments that are grouped in one large increment.

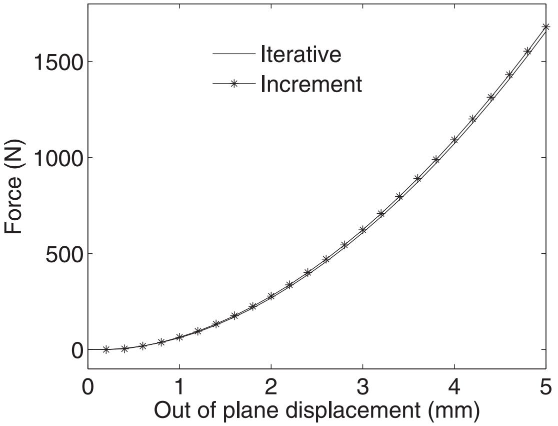

Figure 3.4: A comparison between the in-plane force achieved by the incremental update and the standard Newton approach (iterative) for an increment size of 0.2 mm.

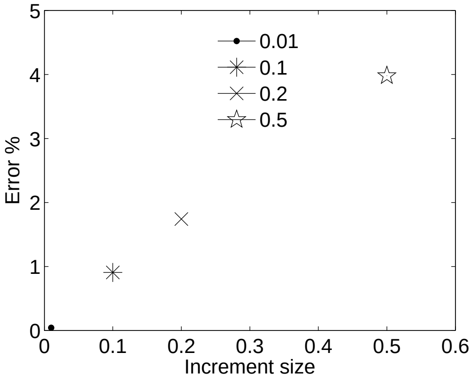

Figure 3.5: The error in reaction force introduced by the incremental compared to the classical implicit approach.

In general, The FE mesh of an efficient implicit SPIF simulation is made of super elements with different update strategy. The classification of these super elements and location is explained in the next section.

3.2 Indicators for nonlinearity

The task in this section is to generically define the update frequency of a super element (iterative, incremental or multi-incremental). This is performed by introducing indicators that can define the super elements classification prior to load increment. These indicators are based and developed for localized plastic deformation. The indicators are the current tool location, the plastic deformation in the previous load increment and the shape change in the previous load increment. The motivation for the indicators and their performance are the topics of the following sections.

3.2.1 Tool indicator

In localized plastic deformation like e.g. SPIF, plastic deformation is introduced in the vicinity of the forming tool. The influence of the tool diameter in the SPIF process is reported in the literature. A smaller tool diameter concentrates the strain under the forming tool and the increase of tool diameter tends to distribute the strain over a more extended area. As the forming tool diameter increases the process becomes more similar to traditional stamping (Jeswiet *et al.*, 2005). Practically, the decrease of the tool diameter increases the forming limits (Fratini *et al.*, 2004). Hirt *et al.* (2002) observes that by decreasing the tool diameter from 30 mm to 6 mm a higher strain and deformation can be achieved.

It becomes intuitive to use the location of the forming tool as an indicator. The tool triggers the location of plastic deformation but it does not define the size of the plastically deforming part in a FE mesh. The size of the plastic region can be defined by finding the super elements that are located in the vicinity of the forming tool. In order to implement that, a number of virtual cross points are introduced in the FE mesh. A cross point can be a common node between 4 adjacent super elements, considering the super elements partitioning in Figure 3.1. Each super element is attached to at least one cross point. The projected distance of the cross points to the tool surface are calculated prior to the load increment. The closest cross point, for instance, to the tool surface can be used to notify that the attached 4 super elements define the plastic region. These super elements have to be updated iteratively in order to capture the introduced plastic deformation. Actually, the cross points are attached to nodes in the FE mesh and as a result the coordinates of the cross points are updated for each load increment and consequently the projected distance to the tool surface. The movement of the tool changes the active cross point(s) and that defines a new set of super elements to be iteratively updated or may be the same set for a number of load increments. In the same way, the rest of the super elements can be classified to use the incremental update or the multi-incremental update strategy.

To investigate the performance of the tool indicator, the test with plastic loading of a blank followed by elastic unloading (penetration test), introduced in 2.2.1, is used. The predicted tool force is used as a parameter to check the accuracy. In this test, four simulations are performed each with a different setting for the tool indicator. These settings define only the closest cross point and cross points within 0.5, 1.0, 1.5 times the tool radius away from the tool surface. The active cross point(s) classify the attached super elements for iterative treatment. The rest of super elements within the FE mesh are classified for incremental update treatment. The predicted force–displacement curves are shown in Figure 3.6. In general, all tool indicator settings successfully predict the pattern of the force–displacement

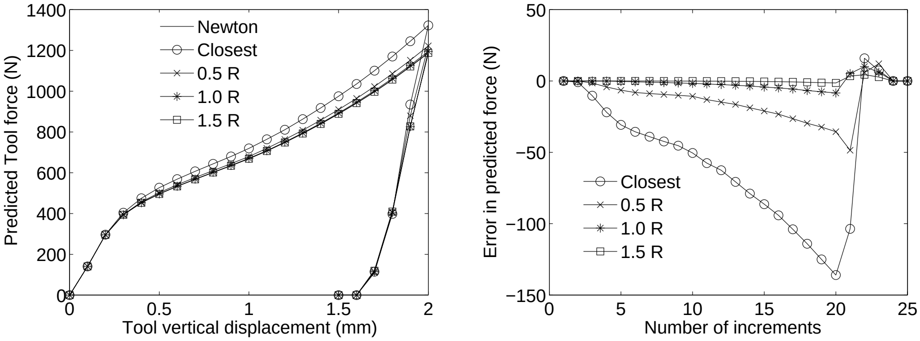

Figure 3.6: The predicted force displacement curve in penetration test test by standard method and iterative–incremental treatment with different tool settings for the tool indicator (left), error evolution during the simulation (right).

Table 3.1: The iterative and the incremental ratios for different tool indicator settings.

	Newton	Closest	0.5 R	1.0 R	1.5 R
α_{Iter} %	100	4	12	16	24
α_{Incr} %	0	96	88	84	76

curve. A larger size of searching radius gives a better prediction of the tool force because it includes more of the plastic region. The result for a searching radius of 1.5 times the tool radius (1.5R) is the best among the tested settings of the tool indicator with maximum error of 4.4 N.

This good result by the 1.5R simulation is achieved because almost the correct size of the plastically deforming part of the FE mesh is updated iteratively. An example of the assumed plastic region and consequently the iteratively treated super elements for 2 different settings of the tool indicator is shown in Figure 3.7. It is observed that the iterative region size and location have been fixed for each tool indicator setting because only a vertical movement has been introduced in this test. The iterative ratio α_{Iter} defines the ratio of iteratively treated super elements to the total number of super element within a FE mesh. Similarly, the incremental ratio α_{Incr} and the multi-incremental ratio α_{MIncr} defines the ratio of incrementally and multi-incrementally treated super elements to the total super elements in a FE mesh respectively. The ratio of each treatment for different tool indicator settings is listed in Table 3.1. Noting that in these simulations the multi-incremental update treatment is not used just to demonstrate a simple classification of the super elements. The computing time of all simulations is summarized in Table 3.2. As expected, the use of one cross point

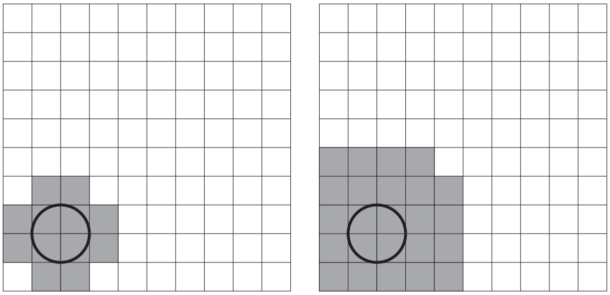

Figure 3.7: The assumed plastic region size and location for 2 different tool indicator settings that are a search radius of 0.5 (left) and 1.5 times the tool radius (right) and the corresponding super element classification into iterative update (gray) and incremental treatment (white).

Table 3.2: The overall computing time and the corresponding speeding factor for the simulations with different tool indicator settings.

	Newton	Closest	0.5 R	1.0 R	1.5 R
CPU time (s)	178.0	85.9	98.8	101.7	109.3
Speeding factor	1.00	2.07	1.80	1.75	1.62

to classify the iterative super elements requires the smallest CPU time of 85.9 s and it accelerates the standard Newton simulation by 2.07 times.

3.2.2 Plastic indicator

The major drawback of the tool indicator is that the searching radius has to be increased in order to define the correct plastic region. This increase of the searching radius may include super elements that do not need to be treated iteratively. Beside, a definition of the proper value of the searching radius requires some experience. For this reason, an indicator that depends on the evolution of displacement field or other derived quantities is more robust. The material nonlinearity is evaluated for each iteration or increment(s) depending on the update strategy even though the incremental and the multi-incremental treated super elements are assumed to model elastic deformation. This means that the plasticity of the previous load increment is known prior to the iterative procedure to solve the system of equations for both the iteratively and the incrementally treated super elements and for the multi-incrementally super elements if they are updated at the previous load increment. This knowledge is used to develop the plastic indicator.

The plastic indicator is based on checking an introduced plastic increment in the previous

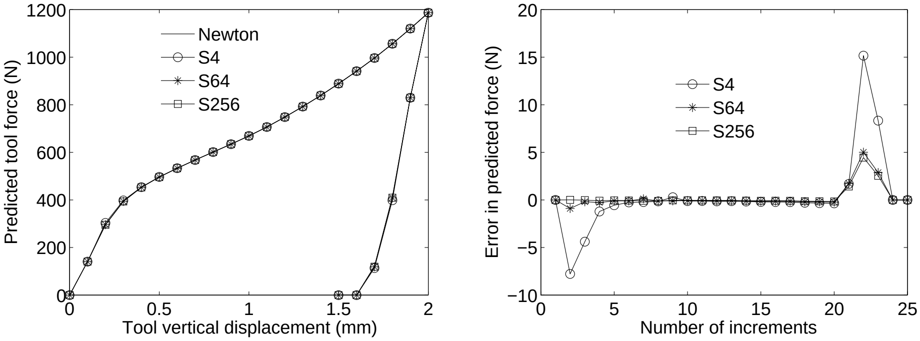

Figure 3.8: The predicted force displacement curve in penetration test by standard method and iterative–incremental treatment with plastic indicator using different super element size (left), error evolution during the simulation (right).

load increment for the iteratively updated and incrementally updated super elements. It checks each integration point of each element within the super element. Based on this check, if at least one integration point within an element indicates a plastic deformation, in the previous load increment, the entire super element is classified for iterative update treatment. Otherwise, the super element is classified for the incremental update treatment. The plastic indicator can generically classify the super elements of a FE mesh into the iterative and incremental update treatment. One integration point can shift a super element from incremental treatment to iterative treatment and vice versa. The influence of the super element size (number of element per super element) becomes more pronounced in the iterative ratio and the increment ratio and consequently on the speeding of the standard implicit simulation.

The performance of the plastic indicator is checked using the penetration test. Three different sizes of the super elements are checked. The different sizes are 4, 64 and 256 elements per super element and they are referred to as S4, S64 and S256. It corresponds to group the FE mesh into 1600, 100 and 25 super elements, respectively. The predicted tool force is shown in Figure 3.8. The largest super element size setting S256 predicts the best fit to the achieved force prediction by the standard Newton implicit simulation. All plastic indicator settings performed very good during the loading stage (till increment number 20) almost zero error except S4 at the first four increments. S4 describes the previous plastic deformation well, but it does not extend to the neighboring super elements that are going to experience plastic deformation in the coming load increment. A larger super element includes more elements that may be treated inefficiently by the iterative update but it is ready to capture the introduction of plastic deformation in the next load increments.

By using the plastic indicator, the super elements are classified into either the iterative or the incremental update treatment. The evolution of the iterative ratio for different super element sizes is shown in Figure 3.9. Regardless of the super element size, the iterative

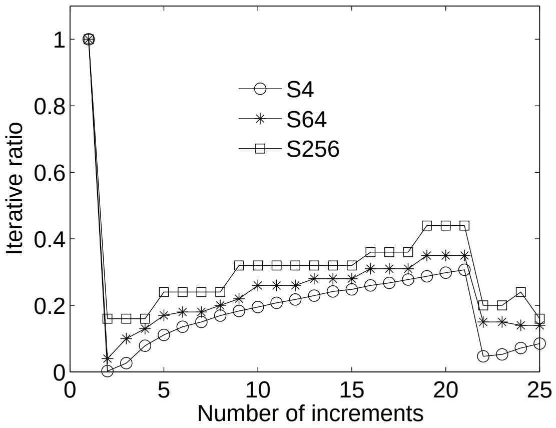

Figure 3.9: Iterative ratio evolution for differrent super element size.

ratio starts at one because the first load increment is used to initialize the FE model and therefore all super elements are set into the iterative treatment. After that, the plastic indicator becomes active and the iterative ratio drops to a low value because the process is still in its initial stage. Then the iterative ratio increases with further plastic deformation that extends over a larger area. During the unloading stage (increment 21 to increment 25), the iterative ratio is expected to reduce to zero, but it is not, and that indicates that a small material nonlinearity occurs during the unloading stage. The difference between the iterative ratios is a result of different super element sizes, a larger super element size predicts a larger iterative ratio. Excluding the first increment, the maximum iterative ratio is 30.7%, 35.0% and 44.0% for S4, S64 and S256, respectively. No super elements are classified into the multi-incremental update treatment therefore the multi-incremental ratio is zero for all simulations. The incremental ratio becomes $\alpha_{\text{Incr}} = 1 - \alpha_{\text{Iter}}$ since the sum of the ratio has to be one.

The plastic indicator successfully defines the location of the plastic deformation zone but the size is influenced by the size of the super element. The largest size of the plastic deformation is defined by the S256 simulation and the smallest by the S4 simulation, a sample of the plastic deformation region at the end of the loading stage is shown in Figure 3.10. The CPU time of the simulations is summarized in Table 3.3. The smaller size of the super element, iterative ratio, is better in accelerating the standard Newton simulation that is speeded up 1.65 times by S4.

3.2.3 Geometrical indicator

In the previous sections, the super elements are classified into iterative and incremental update treatment only and the multi-incremental treatment is not included in the performed

Table 3.3: The overall computing time and the corresponding speeding factor using the plastic indicator.

	Newton	S4	S64	S256
CPU time (s)	178.0	107.7	110.1	116.1
Speeding factor	1.00	1.65	1.62	1.53

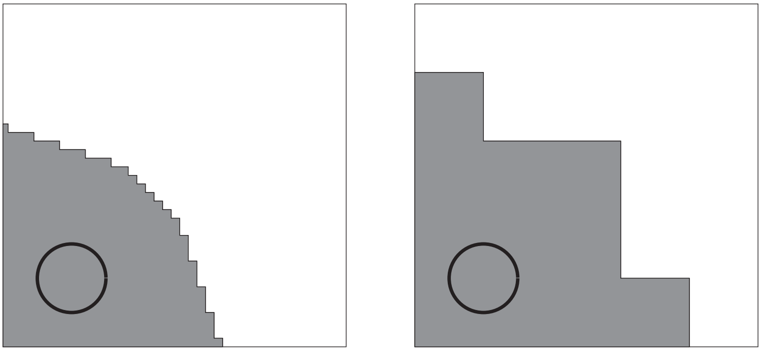

Figure 3.10: Predicted plastic region by differrent super element size: S4 (left) and S256 (right).

simulations. The tool indicator can be extended to classify the super elements into three different update frequency regions by introducing two searching radii. One of these radii can classify the super elements at a given distance from the tool surface into the iterative treatment while a larger searching radius classifies the neighbors (to some extent) into the incremental update treatment. The rest of super elements are set into the multi-incremental update treatment. By this setting, the movement of the tool results in a change of the super elements classifications. A formerly multi-incremental super element is updated if it is located in the range of the incremental searching radius and it will be updated each increment afterward, otherwise it is not updated and that may be for the entire simulation. Another option for updating a multi-incremental super element is to perform the update after a number of increments. This option is more dependent on the expertise of the analyst, but it does not consider the influence of shape change. For this reason, a geometrically based indicator is developed to control the frequency of updating a multi-incremental treated super element.

Super elements based simulation controls the frequency of updating super elements. Meanwhile the system of equations, including all degrees of freedom, are solved in each iteration and that updates the incremental displacement vector d. Consequently, the coordinates in the model are updated and that results in a shape change in the entire FE mesh. Within the structure of a multi-incremental treated super element, a check is performed

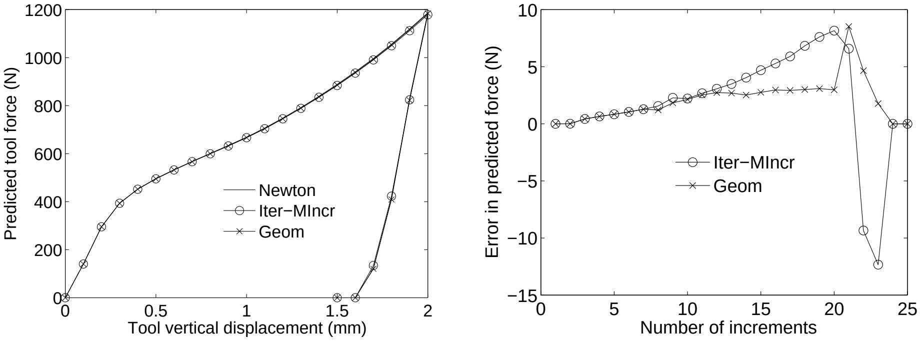

Figure 3.11: The predicted force displacement curve in penetration test by standard method and super element method based on iterative and multi-incremental treatments with and without geometrical indicator (left), error evolution during the simulation (right).

that calculates the normal vector of each element within the super element. This check is performed incrementally after the convergence of the iterative procedure. The newly calculated element normal vector is compared to the calculated normal vector at the moment of updating the geometrical and the material nonlinearities before several increments. The normal vector change in orientation by an angle θ can be used to indicate a shape change. Based on that, the change of only one element can trigger that the entire super element requires an update. The update is performed for that particular multi-incremental super element prior to the new load increment and a new update classification can be assigned to the super element.

The penetration test is used to check the performance of the geometrical indicator. Two tests are performed and compared to the standard Newton simulation. The first simulation has either iterative or multi-incremental treated super elements (Iter–MIncr), no incremental region. Based on the tool, a search radius is used to classify a group of super elements into the iterative treatment. The classification of the super elements is fixed and the multi-incremental super elements are not updated for the entire simulation. The second test has the same setting as the first test and the geometrical indicator is activated (Geom). A change in an element normal vector orientation by half a degree is used as threshold to turn a multi-incremental treated super element into an incremental treated super element. The predicted force is shown in Figure 3.11. The introduction of the geometrical indicator reduces the error in the force prediction to less than 9 N. The indicator helps in enhancing the prediction of the force compared to the simulation without the use of the geometrical indicator.

The evolution of the super element classification ratios (iterative, incremental and multi-incremental) is shown in Figure 3.12. In both simulations, all super elements are classified as iterative treatment for the first load increment and for the rest of the simulation the indicator always results in an iterative ratio of 24%. After increment 7, the geometrical indicator successfuly starts changing some of multi-incremental super elements into the

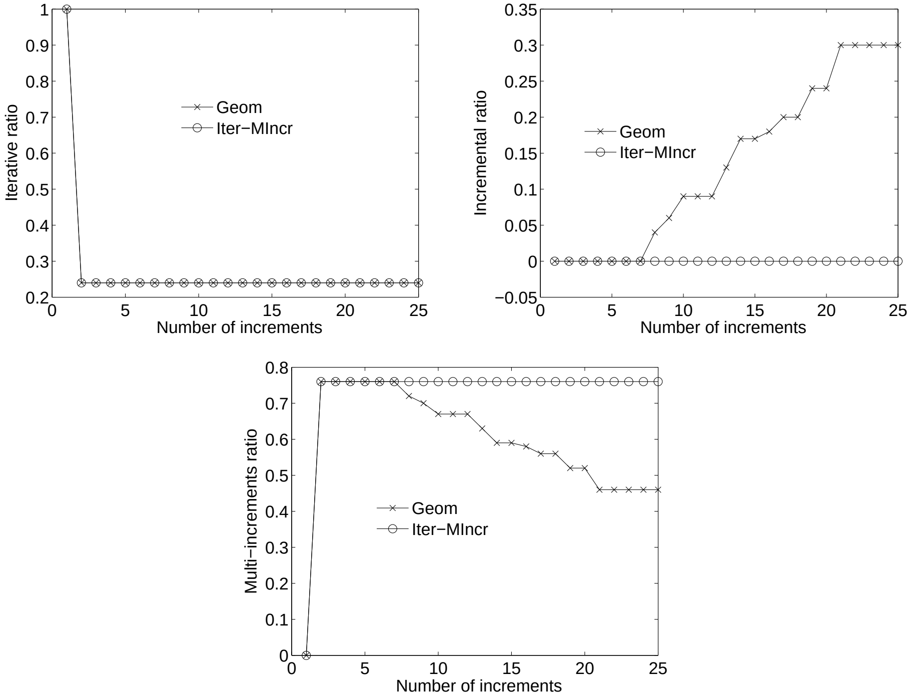

Figure 3.12: The evolution of the super element classifications into iterative (top left), incremental (top right) and multi-incremental (bottom).

Table 3.4: The overall computing time and the corresponding speeding factor using the plastic indicator.

	Newton	Geom	Iter–MIncr
CPU time (s)	178.0	89.1	84.6
Speeding factor	1.00	2.00	2.10

incremental treatment, therefore the incremental ratio is increased from zero to 30% and the multi-incremental ratio is reduced by the same value from 76% to 46%. The simulation using the geometrical indicator requires slightly more CPU time than the simulation without the indicator but it is still twice as fast as the standard Newton simulation, the performance of the simulations is listed in Table 3.4.

3.3 Case studies

Up to this point, the efficient implicit method based on super elements is introduced and tested using a simple test of localised deformation. Also, several indicators are introduced to classify the super elements into an iterative, incremental and multi-incremental updated strategy. Now, two real-life applications are used to demonstrate the capability of the efficient implicit method with the use of the indicators to speed up the standard Newton implicit simulation of these applications. The first case study is performing one loop of single point incremental forming SPIF and the second case is to perform continuous bending under tension of a strip by three rolls.

3.3.1 One loop of SPIF

In this case study, one loop of a single point incremental forming SPIF process is simulated. The numerical model that is introduced in Section 2.2.1 is used here. The simulation starts by moving the tool that is initially in contact 0.5 mm downwards in the vertical direction (z coordinate) to introduce a penetration in the blank. At this vertical position, the forming tool starts moving in–plane following a tool path that draws a rectangle in the blank and ends at the starting position of in–plane movement. The initial tool position and the tool path are shown in Figure 3.13.

Two simulations are performed based on the efficient implicit approach. In both simulations, the FE mesh is grouped into 1600 super elements using super elements that include 4 triangular shell elements. The first simulation uses only iterative and incremental updated super elements (Iter–Incr). The tool indicator is used to classify the super elements within a distance of one radius of the tool from the tool surface into iterative treatment. The plastic indicator is also used. The second simulation uses all types of treatment (All type). A search radius classifies the super elements within a distance of four times the tool radius from the tool surface into the incremental treatment then the search radius of the iterative treatment, that is as large as one radius of the tool, is applied. The plastic indicator is active and a geometrical indicator with half a degree threshold is used to shift multi-incremental super element into incremental super element.

The achieved xz profile at $y = 0$ is used to compare the performance of the efficient implicit simulations to the standard Newton simulation after one loop of the SPIF process, the profiles are shown in Figure 3.14. Both simulations predict a very good result, the error in predicting the profile is limited to less than $0.4\ \mu$m by the Iter–Incr simulation while it is less than $2\ \mu$m by the second simulation. A slight deviation between the Iter–Incr and the second simulation is observed because part of the FE model is multi-incrementally treated.

In both simulations, the indicators successfully classify the super elements into the available update treatments, Figure 3.15. As the tool approaches the corner of the rectangular shape, more plastic deformation is introduced; consequently an increase of the iterative ratio is observed because the corners are stiffer (double clamped edges) than the middle position between two corners. The movement of the tool away from the corner results in relaxing (local springback) of a plastically deformed part and that results in a drop of the iterative ratio. This is observered for corners $2, 3$ and 4. Approaching corner 1 at the end of the tool path shows a reduction in the iterative ratio because the region has been deformed plastically

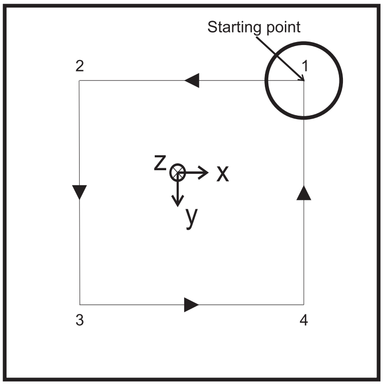

Figure 3.13: One loop tool path for SPIF simulation. The tool path length is 260.5 mm that consists of 0.5 mm vertical movement followed by 260 mm in–plane movement.

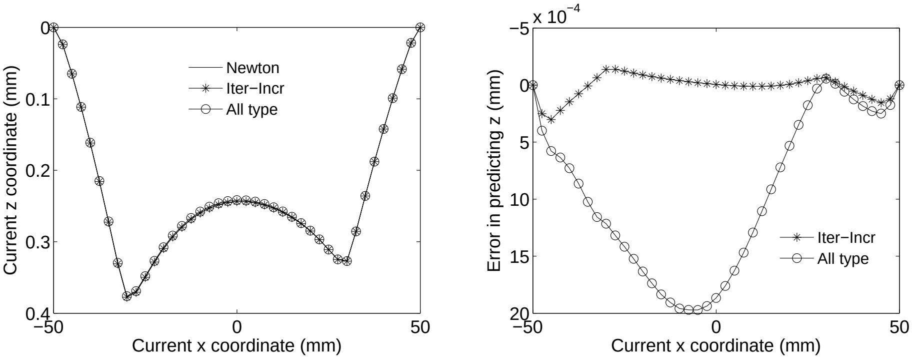

Figure 3.14: The numerical xz profile at $y = 0$ at the end of one loop SPIF process (left) and the error by the efficient implicit for two different settings.

before. The incremental ratio shows the opposite behaviour in the Iter–Incr simulation since the sum of both ratio has to be one. In the second simulation, the incremental ratio starts at low level because the rest of the super elements are classified to the multi-incremental treatment. After that, it shows a significant increase hence more of the multi-incremental super elements are turned into the incremental treatment and this lasts until increment 614 when all of the multi-incremental super elements are gradually turned into the incremental treatment. Now, the super elements are either iteratively or incrementally treated. The number of multi-incremental super elements always decreases because no criterion is used in this simulation to classify an iterative or an incrementally treated super element into

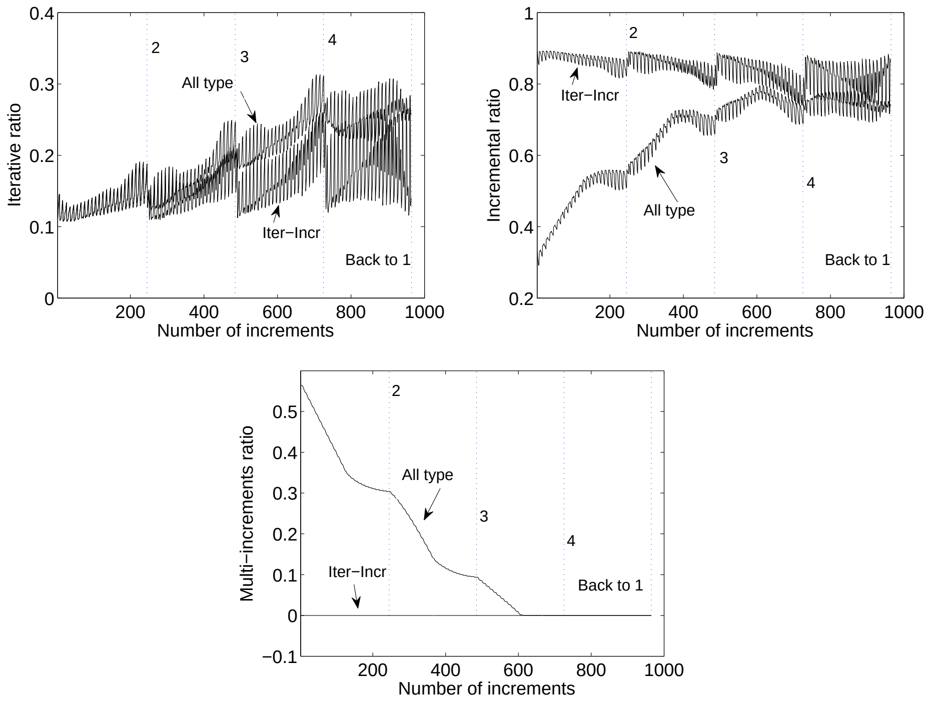

Figure 3.15: Super element classifications evolution during the simulations into iterative (top left), incremental (top right) and multi-incremental treatment (bottom) .

multi-incremental treatment.

The CPU time required by the simulations is listed in Table 3.5. The standard Newton simulation requires 5693.2 s (1.58 hr) to finish 965 load increments. The Iter–Incr simulation performs the same number of load increments in 3393.2 s (0.94 hr) that is 1.68 times faster than the standard Newton simulation and less than 0.4 μm error is observed in the predicted xz profile. The simulation with all indicators active is slightly faster, it is 1.72 times faster than the standard simulation but with 2 μm error.

3.3.2 Continuous bending under tension

In this test, a strip is deformed by continuous bending under tension. The deformation is introduced by the use of a roll set made of three identical rolls of 15 mm diameter. The rolls are separated from each other by 17.5 mm, in longitudinal direction. The strip dimensions are 200 × 20 × 1 mm^3. A 2 D model of the strip and the roll set is shown in Figure 3.16. The process is started by clamping both ends of the strip. Then the strip is bent by the movement of the central (upper) roll 3.3 mm downward. Meanwhile, the two lower rolls are fixed in

Table 3.5: The overall computing time and the corresponding speeding factor for one loop of SPIF simulation.

	Newton	Iter–Incr	All type
CPU time (s)	5693.7	3393.2	3306.5
Speeding factor	1.00	1.68	1.72

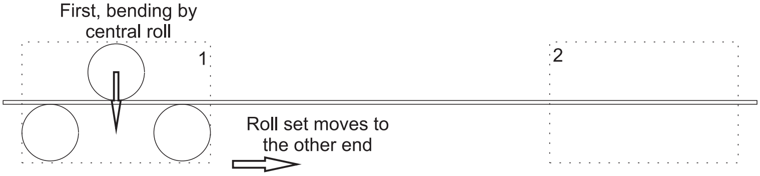
First, bending by central roll
1
2
Roll set moves to the other end

Figure 3.16: 2 D presentation of the strip and the roll set, continuous bending under tension precess description.

position, this is the first stage of the process. The process proceeds by moving the roll set to the opposite end of the strip (second stage). This process can be classified as multi–point incremental forming.

Half of the strip, taking advantage of symmetry around the longitudinal axis, is modeled by 4000 triangular shell elements. The material model is representative for mild steel. The anisotropic yield behaviour of the material is modelled by the Hill'48 criterion. The isotropic nonlinear hardening is governed by the power law as

$$\sigma = 494(\varepsilon + 0.00001)^{0.248} \tag{3.5}$$

where σ and ε are the flow stress and the equivalent plastic strain respectively. The material has a Young's modulus of 200 GPa and Poisson's ratio of 0.3.

An efficient implicit simulation based on super element is performed and achieved results are compared to the results of the standard Newton simulation. For efficient implicit simulation, the FE mesh is grouped into 1000 super elements and they are classified into iterative or incremental update treatment only (Iter–Incr). The predicted vertical force at the central roll, for instance, is plotted in Figure 3.17. The strip experiences bending by the vertical movement of the upper roll combined with tension because of the clamped ends. For that, the force on the roll is increased monotonically till the end of this stage. A global plastic deformation is observed in the entire strip. Parts of the strip that are in the vicinity of the rolls harden more than the rest of the strip that experiences mainly tension. By the roll set horizontal movement, a new part of the strip is plastically deforming and that reduces the tension in the strip and consequently the predicted vertical force at the upper roll. An increase of the predicted force is observed because the material that is hardened at the lower

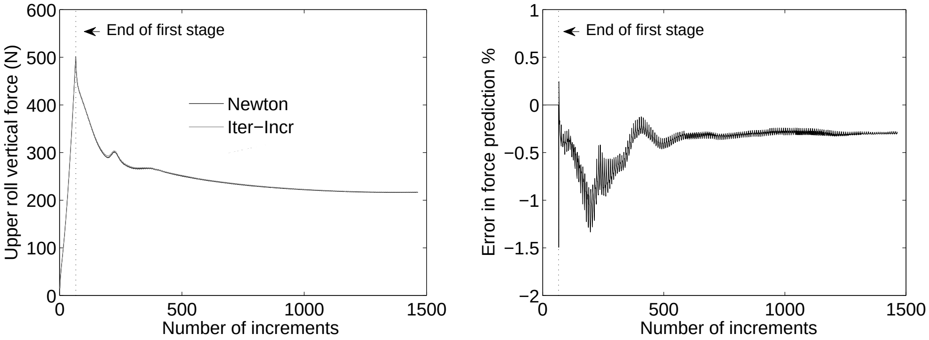

Figure 3.17: Predicted vertical force at the central roll (left) and error by the Iter–Incr based simulation (right).

right roll during the first stage passes the upper (central) roll where it experiences a harder material. The result achieved with Iter–Incr update strategy has a very good agreement with the standard Newton result with less than 1.5% error

The Iter–Incr simulation has zero error at the first stage of the simulation since all super elements are treated iteratively. This iterative treatment is necessary because of the global plastic deformation. During the second stage, the tool indicator is used with a search radius that classifies super elements within 5 mm of any roll surface into iterative treatment. The evolution of the iterative and the incremental ratio are shown in Figure 3.18. This results in almost constant value for the iterative and incremental ratio of 0.224 and 0.776, respectively. Of course, the positions of the iterative treated super elements are updated to be in the vicinity of the rolls.

Based on this setting, the Iter–Incr implicit simulation is faster by 1.81 times than the standard Newton implicit simulation. The Iter–Incr simulation requires 9075.2 s (2.52 hr) to perform 1466 load increments, while the standard Newtom simulation requires 16385.2 s (4.55 hr) for the same number of increments.

3.4 Summary and conclusions

In Section 3.1, a super element based efficient implicit time integration procedure is introduced. A FE mesh is substructured into super elements. This re–cast the assembly procedure to be over all super elements instead of elements. It facilitates partitioning a FE mesh into different update frequencies: each iteration, increment and multi-increments. Also, it manages and organizes the update of the tangent stiffness matrix and the internal force vector based on the super element classified treatment. A simple test is used to investigate the influence of linearization of geometrical nonlinearity within the incremental treatment. Under the condition of small incremental deformation, the incrementally linearized approach can be considered valid.

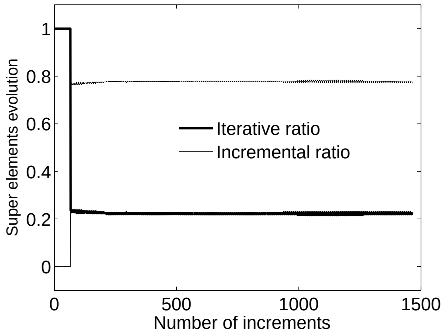

Figure 3.18: Super element evolution during the Iter–Incr simulation.

In Section 3.2, super elements are generically classified into different update frequency strategies (iterative, incremental or multi-incremental). This is performed by introducing indicators that can define the super elements classification prior to load increment. These indicators are developed for localized plastic deformation. The indicators are the current tool location, plastic deformation in the previous load increment and the shape change in the previous load increment. The tool indicator uses the strategy of a search radius to classify the super elements based on the distance between the super element and the tool surface. The plastic deformation in the previous load increment is used in the plastic indicator in order to turn incrementally treated super elements into iterative treatment. The geometrical indicator that is based on shape change classifies multi-incremental super elements into incremental treatment. In general, all indicators perform very good and predict the necessary change in the super element treatments.

The super element based efficient implicit approach and the developed indicators are tested by simulating two practical applications in order to demonstrate the capability of speeding up the standard Newton implicit simulation, in section Section 3.3. It speeds up one loop of SPIF simulation by 1.68 times with error limited to $0.4\,\mu$m in the predicted xz profile. A speeding factor of 1.81 is obtained in the simulation of the deformation of a strip by continuous bending under tension. During the entire simulation, the error of predicting the upper roll vertical force does not exceed 1.5%.

In this chapter, many simulations are performed by the super element based efficient implicit approach and a speeding factor in the range of $1.5 - 2.1$ is achieved. However, two simulations have to be performed, a standard and a super element based, in order to know the speeding factor. This brings to attention the need for a model that predicts in advance the expected speeding factor and to define the limits of such a speeding factor. This will be the topic of the next chapter.

4. Analysis of the speeding factor

Up to this point, the super element efficient implicit procedure successfully accelerates the implicit simulation of localised deformation. The achieved speeding up of the implicit simulation is simply explained by reducing the cost of the increment. This is applied by efficient update strategies that distinguish between plastic and elastic deformation. A better understanding of the computational performance of the super element implicit procedure is motivated by several issues in Section 4.1. For instance, a change in the used number of iterations or the combination of the update frequencies can have an impact on the achieved speeding of the standard implicit simulation. Clearly, there is need for an analytical formula to explain an achieved speeding factor and even to predict in advance the speeding factor for an implicit simulation. Also, the analytical formula can provide a guideline for optimum performance. An analytical formula for the two domain approach is presented in Section 4.2, it focuses on the performance of the speeding factor for a FE model assembled of iterative and incremental updated super elements. This analytical formula is extended in Section 4.3 to include the influence of introducing a third domain that has low update frequency (multi-incremental). The achieved speeding factor in one loop efficient implicit simulation of SPIF process is explained as demonstration in Section 4.4.

4.1 Motivation

The speeding factor is defined as the ratio of CPU time used by the standand Newton implicit simulation to the efficient implicit simulation time. It is used to measure the advantage of the efficient implicit approach in accelerating the standard implicit simulation. It has to be larger than one to be successful. The basic idea of the efficient implicit procedure is to reduce the cost of each iteration during the iterative procedure, as explained in Section 2.3. The reduction in iteration cost is a result of reducing the cost of building the system of equations (BUILD) and updating the stresses based on the actual displacement (UPDATE). BUILD and UPDATE are two major costs of the iteration out of three. The third major cost is solving the system of equations (SOLVE) and this part of the iteration cost is not influenced by the efficient implicit procedure. An example of these costs is shown in Figure 2.14.

The reduction in the cost of BUILD and UPDATE is strongly influenced by the update frequency ratios (iterative, incremental and multi-incremental) that are used in a FE analysis. The iterative treatment of all super elements in a FE model results in a speeding factor of one, the implicit simulation in this case is not accelerated. For a number of iterations within one increment, the incremental update treatment performs BUILD and UPDATE only once per increment (in the first iteration). The redundant cost of BUILD and UPDATE for the rest of the iterations is saved. This results in a speeding factor higher than one. Intuitively, the multi-incremental treatment of the entire FE model has a larger impact on the speeding factor than the incremental treatment. It has the same update strategy as the incremental method but extended for a number of increments where the incremental treatment forces the update every increment. In the efficient implicit simulation, the FE models may consist of the three types of updating strategy. The presence of the iterative update strategy is essential to model the plastic localised deformation

A different material model or element type may increase or decrease the required CPU time by BUILD and UPDATE and a new speeding factor will be achieved. To conclude, the speeding factor of an implicit simulation depends on several factors that are the update strategies ratio in a FE model, the frequency of performing the update, the number of iterations that are performed for a number of increments and the major cost ratios of the Newton iteration. The dependency of the speeding factor on these factors motivates the development of an analytical formula that estimates in advance the speeding factor for a FE model implicit simulation.

4.2 Two domain analytical formula

In this section, an analytical formula is introduced to predict the speeding factor that the two domain efficient implicit method can achieve compared to the standard Newton implicit simulation of a FE model. The basic idea of two domain method is introduced in Section 2.3.2 and the super element based implementation is explained in Section 3.1, The two domain method splits a FE model into an iterative domain (super elements) and an incremental domain. An iterative updated domain is recommended for plastic deformation while an incremental domain is suitable for elastic deformation. The cost of one Newton iteration (T) can be split into three major parts that are: building the system of equations (B), solving the system of equations (S) and updating the stresses based on actual displacement (U)[1]

$$T = B + S + U \tag{4.1}$$

The cost ratio of the major parts is defined as the ratio of each partial cost (CPU time) to the total time of the iteration

$$\frac{B + S + U}{T} = B_{\mathrm{T}} + S_{\mathrm{T}} + U_{\mathrm{T}} = 1 \tag{4.2}$$

where B_{T}, S_{T} and U_{T} are the cost ratio of BUILD, SOLVE and UPDATE respectively.

The cost of building the system of equations consists of the cost of creating the tangent stiffness matrix and the internal force vector. It can be defined as

$$B = \alpha_{\mathrm{P}} B^{\mathrm{P}} + \alpha_{\mathrm{E}} B^{\mathrm{E}} \tag{4.3}$$

where B^{P} and B^{E} are the cost of building the entire system of equations as if it deforms plastically or elastically, respectively. α_{P} and α_{E} define the ratio of plastic and elastic elements to the total number of elements in a FE mesh. For simplification, an assumption

[1] The first letter of the abbreviated terminology will be used in the equations for convenience

is made that an element is either plastically or elastically deforming[2]. The cost of building a system of equations for a plastically deforming model is larger than for an elastically deforming model. The tangent stiffness matrix for elastically loading or unloading elements is based on the elasticity tensor while for plastically deforming elements it is based on the tensor of tangent elastoplastic moduli (Simo and Hughes, 2000). It consists of the elasticity tensor and an additional term that depends on the used material model. The elastic/plastic BUILD ratio β is defined as the ratio of the CPU cost of building an elastic system of equations to the cost of building a plastic system of equations

$$\beta = \frac{B^{\mathrm{E}}}{B^{\mathrm{P}}} \tag{4.4}$$

by substituting (4.4) into (4.3) and $\alpha_{\mathrm{P}} + \alpha_{\mathrm{E}} = 1$, B can be presented as a function of B^{P} or B^{E}

$$B = \big(\alpha_{\mathrm{P}} + (1-\alpha_{\mathrm{P}})\beta\big)B^{\mathrm{P}} = \Big(\frac{\alpha_{\mathrm{E}}\beta + 1 - \alpha_{\mathrm{E}}}{\beta}\Big)B^{\mathrm{E}} \tag{4.5}$$

Similarly, the cost of updating the stresses based on actual displacement can be defined

$$U = \alpha_{\mathrm{P}} U^{\mathrm{P}} + \alpha_{\mathrm{E}} U^{\mathrm{E}} \tag{4.6}$$

The actual stresses are updated by an iterative procedure that is known as return mapping algorithm. It finds the balance between the elastic and the plastic strains. At the beginning of the procedure, a trial elastic stress estimation is made to decide if the stresses are developed elastically or plastically based on a defined yield condition (Simo and Hughes, 2000). If the yield condition is not violated the procedure finishes and the stress is updated elastically. Otherwise, an iterative procedure starts to find the balance between the elastic and the plastic strains and the corresponding stresses. The procedure is performed for each integration point. The cost of the iterative procedure is strongly dependent on the complexity of the material model. More time is required to perform a fully plastic update of the stresses U^{P} compared to a fully elastic update of the stresses U^{E}. The elastic/plastic UPDATE ratio η is defined as the ratio of the cost for updating the stresses in the elastic case to updating the stresses for the plastic case

$$\eta = \frac{U^{\mathrm{E}}}{U^{\mathrm{P}}} \tag{4.7}$$

then, U can be defined as

$$U = \big(\alpha_{\mathrm{P}} + (1-\alpha_{\mathrm{P}})\eta\big)U^{\mathrm{P}} = \Big(\frac{\alpha_{\mathrm{E}}\eta + 1 - \alpha_{\mathrm{E}}}{\eta}\Big)U^{\mathrm{E}} \tag{4.8}$$

Now, the cost of one increment of the two domain method can be split into three parts. These parts are: the cost of the iterative domain, the cost of the incremental domain and the cost of solving the system of equations. The cost of the iterative domain (CITER) is the cost of building the system of equations and updating the stresses of the iterative super elements. It is required for every iteration

$$\mathrm{CITER} = \alpha_{\mathrm{Iter}}(B^{\mathrm{P}} + U^{\mathrm{P}})N \tag{4.9}$$

[2] In general, an element can have both elastic and plastic integration points

where N is the number of iterations and α_{Iter} is the iterative ratio that is defined in Section 3.2.1. The iterative ratio should approximately equal the ratio of plastic elements to the total elements in the FE model $\alpha_{\text{P}} = \alpha_{\text{Iter}}$. With the use of (4.5) and (4.8), CITER becomes

$$\text{CITER} = \alpha_{\text{Iter}}\Big(\frac{B}{\alpha_{\text{Iter}} + (1 - \alpha_{\text{Iter}})\beta} + \frac{U}{\alpha_{\text{Iter}} + (1 - \alpha_{\text{Iter}})\eta}\Big)N \tag{4.10}$$

The cost of the incremental domain (CINCR) is the cost of building the system of equations and the update of the stresses (nonlinearly) only once at the beginning of the increment. Within the increment, the internal force vector is updated linearly by the multiplication of the super element tangent stiffness matrix and the super element incremental displacement which requires negligible cost compared to an update based on the actual stresses. Assuming that the incremental ratio (Section 3.2.1) is equal to the ratio of elastic elements to the total elements in the FE model $\alpha_{\text{E}} = \alpha_{\text{Incr}}$, CINCR equals

$$\begin{aligned} \text{CINCR} &= \alpha_{\text{Incr}}(B^{\text{E}} + U^{\text{E}}) \\ &= \alpha_{\text{Incr}}\Big(\frac{\beta B}{\beta\alpha_{\text{Incr}} - \alpha_{\text{Incr}} + 1} + \frac{\eta U}{\eta\alpha_{\text{Incr}} - \alpha_{\text{Incr}} + 1}\Big) \end{aligned} \tag{4.11}$$

The cost of solving the system of equations in the two domain method is similar to the cost in the standard Newton method because the size of the system of equations is not changed.

Now, the speeding factor (SPEED) is defined as the ratio of one Newton increment cost to the cost of one increment of the two domain method

$$\text{SPEED} = \frac{NT}{NS + \text{CITER} + \text{CINCR}} \tag{4.12}$$

dividing over T presents a generalized model of SPEED that depends on the major parts cost ratio instead of the major parts absolute cost

$$\text{SPEED} = \frac{N}{NS_{\text{T}} + \text{CITER}_{\text{T}} + \text{CINCR}_{\text{T}}} \tag{4.13}$$

where

$$\text{CITER}_{\text{T}} = \frac{\text{CITER}}{T} = \alpha_{\text{Iter}}\Big(\frac{B_{\text{T}}}{\alpha_{\text{Iter}} + (1 - \alpha_{\text{Iter}})\beta} + \frac{U_{\text{T}}}{\alpha_{\text{Iter}} + (1 - \alpha_{\text{Iter}})\eta}\Big)N \tag{4.14}$$

$$\text{CINCR}_{\text{T}} = \frac{\text{CINCR}}{T} = \alpha_{\text{Incr}}\Big(\frac{\beta B_{\text{T}}}{\beta\alpha_{\text{Incr}} - \alpha_{\text{Incr}} + 1} + \frac{\eta U_{\text{T}}}{\eta\alpha_{\text{Incr}} - \alpha_{\text{Incr}} + 1}\Big) \tag{4.15}$$

Special cases

Several special cases are introduced here to check the performance of the analytical formula. The first special case is to check the expected SPEED if *one iteration is performed per increment*. Logically, the two domain method does not accelerate the standard Newton implicit simulation for one iteration per increment because the two domain method advantage is to save the cost of BUILD and UPDATE of the elastic part after the first iteration. Substitute

$N = 1$ in (4.13) and either $\alpha_{\text{Iter}} = 1 - \alpha_{\text{Incr}}$ in (4.14) or $\alpha_{\text{Incr}} = 1 - \alpha_{\text{Iter}}$ in (4.15) and manipulate terms

$$\begin{aligned} \text{SPEED} &= \frac{1}{S_{\text{T}} + U_{\text{T}}\Big(\frac{\alpha_{\text{Iter}} + (1-\alpha_{\text{Iter}})\eta}{\alpha_{\text{Iter}} + (1-\alpha_{\text{Iter}})\eta}\Big) + B_{\text{T}}\Big(\frac{\alpha_{\text{Iter}} + (1-\alpha_{\text{Iter}})\beta}{\alpha_{\text{Iter}} + (1-\alpha_{\text{Iter}})\beta}\Big)} \\ &= \frac{1}{\underbrace{S_{\text{T}} + B_{\text{T}} + U_{\text{T}}}_{1}} = 1 \end{aligned} \tag{4.16}$$

The second case is *performing iterative update for all super elements*. This is similar to the standard Newton method, hence SPEED has to be one. Substitute $\alpha_{\text{Incr}} = 0 \Rightarrow \text{CINCR} = 0$ and $\alpha_{\text{Iter}} = 1$ in (4.13)

$$\text{SPEED} = \frac{N}{NS_{\text{T}} + N(B_{\text{T}} + U_{\text{T}})} = \frac{N}{N(S_{\text{T}} + B_{\text{T}} + U_{\text{T}})} = 1 \tag{4.17}$$

The third case is *dominating solver cost*. This means that $S_{\text{T}} \approx 1 \Rightarrow B_{\text{T}} \approx U_{\text{T}} \approx 0$, thus

$$\text{SPEED} = \frac{N}{NS_{\text{T}}} = \frac{1}{S_{\text{T}}} \approx 1 \tag{4.18}$$

The fourth case is *complex material model*, resulting in a negligible elastic/plastic BUILD ratio β and elastic/plastic UPDATE ratio η. $\beta \approx \eta \approx 0 \Rightarrow \text{CINCR} \approx 0$ and consequently SPEED equals one. The fifth case is *negligible solver cost*. $S_{\text{t}} \approx 0 \Rightarrow B_{\text{T}} + U_{\text{T}} \approx 1$, SPEED becomes

$$\text{SPEED} = \frac{N}{\text{CITER}_{\text{T}} + \text{CINCR}_{\text{T}}} \tag{4.19}$$

if $\alpha_{\text{Iter}} = 1 \Rightarrow \text{CINCR}_{\text{T}} = 0$ and $\text{CITER}_{\text{T}} = N$, SPEED becomes equal to one. If $\alpha_{\text{Incr}} = 1 \Rightarrow \text{CITER}_{\text{T}} = 0$, SPEED equals

$$\text{SPEED} = \frac{N}{\underbrace{B_{\text{T}} + U_{\text{T}}}_{1}} = N \tag{4.20}$$

This means that the two domain method can accelerate the standard Newton simulation by a factor equal to the number of iterations used per increment.

Typical case

For moderately localised forming implicit simulation, all the factors of SPEED are involved. SPEED is expected to be in the range of one to the number of iterations per increment N. This hints that a large number of iterations is preferred to have better SPEED. For this reason, the limit of SPEED (4.13) at $N \to \infty$ is calculated (l'Hôpital's rule is applied)

$$\begin{aligned} \lim_{N\to\infty} \text{SPEED} &= \lim_{N\to\infty} \frac{N}{NS_{\text{T}} + \text{CITER}_{\text{T}} + \text{CINCR}_{\text{T}}} \\ &= \frac{1}{S_{\text{T}} + \alpha_{\text{Iter}}\Big(\frac{B_{\text{T}}}{\alpha_{\text{Iter}} + (1-\alpha_{\text{Iter}})\beta} + \frac{U_{\text{T}}}{\alpha_{\text{Iter}} + (1-\alpha_{\text{Iter}})\eta}\Big)} \end{aligned} \tag{4.21}$$

Table 4.1: The settings for different case studies in the evaluation of SPEED.

	α_{Iter}	β	N	S_{T}	
Case 1	variable	1.0	5	variable	Figure 4.1
Case 2	0.1	variable	5	variable	Figure 4.2
Case 3	0.1	1.0	variable	variable	Figure 4.3
Case 4	0.1	1.0	∞	0.1	Figure 4.4
Case 5	variable	1.0	∞	variable	Figure 4.5
Case 6	0.1	variable	∞	variable	Figure 4.6

The upper limit of SPEED is a function of the major parts ratio, iterative ratio, elastic/plastic BUILD ratio β and elastic/plastic UPDATE ratio η. It is independent of CINCR$_{\text{T}}$. A simplified analytical formula of SPEED can be introduced from (4.13) by assuming that $\beta = \eta$ as

$$\text{SPEED} = \frac{N\big(\alpha_{\text{Iter}} + (1-\alpha_{\text{Iter}})\beta\big)}{(1-\alpha_{\text{Iter}})\beta + N\alpha_{\text{Iter}} + \beta S_{\text{T}}\big(N(1-\alpha_{\text{Iter}}) + \alpha_{\text{Iter}} - 1\big)} \tag{4.22}$$

and consequently a simplified upper limit of SPEED becomes

$$\lim_{N\to\infty} \text{SPEED} = \frac{\alpha_{\text{Iter}} + (1-\alpha_{\text{Iter}})\beta}{\alpha_{\text{Iter}} + (1-\alpha_{\text{Iter}})\beta S_{\text{T}}} \tag{4.23}$$

The simplified SPEED model is used for further investigations studying the influence of the iterative ratio α_{Iter}, the elastic/plastic BUILD ratio β and the number of iterations N on the two domain SPEED for the entire range of the partial cost of SOLVE S_{T}. The settings of the studied cases are summarized in Table 4.1. The SPEED curves for different iterative ratio α_{Iter} values are shown in Figure 4.1. An upper limit and a lower limit is achieved by $\alpha_{\text{Iter}} = 0$ and $\alpha_{\text{Iter}} = 1.0$, respectively. The lower limit is not influenced by S_{T} because the entire system is updated each iteration with $\alpha_{\text{Iter}} = 1.0$. For $\alpha_{\text{Iter}} < 1.0$, SPEED performs better at low S_{T} that corresponds to high value of B_{T} and U_{T}. The performance of SPEED is enhanced by reducing α_{Iter}. The best achieved SPEED is at $\alpha_{\text{Iter}} = 0$ that is fully incremental treatment of the entire FE model. The influence of α_{Iter} on SPEED is reduced at higher S_{T} e.g. at $S_{\text{T}} = 0.1$ the increase of α_{Iter} from 0.1 to 0.5 reduces SPEED by 45 % while at $S_{\text{T}} = 0.5$ it reduces SPEED only by 20 %.

The influence of the elastic/plastic BUILD ratio β on SPEED curves is plotted in Figure 4.2. A lower limit is observed at $\beta = 0$ because SPEED becomes dominant by the iterative cost while an upper limit is observed at $\beta = 1.0$ because the elastic cost of BUILD and UPDATE is treated as expensive as the plastic cost (very simple material model). The same trend of SPEED is observed as in Figure 4.1. The increase of β enhances the achieved SPEED. The influence of β on SPEED is larger at low S_{T} than at high S_{T}.

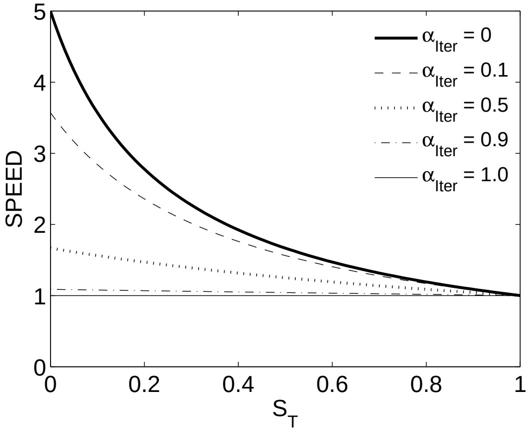

Figure 4.1: SPEED curves for different iterative ratio values, $N = 5$ and $\beta = 1.0$.

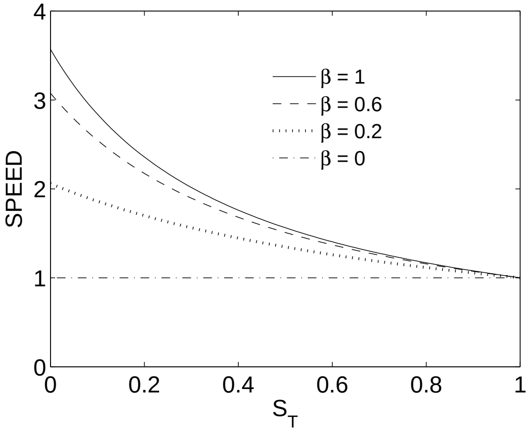

Figure 4.2: SPEED curves for different values of the elastic/plastic BUILD ratio β, $N = 5$ and $\alpha_{\text{Iter}} = 0.1$.

The achieved SPEED curves for different number of iterations N are plotted in Figure 4.3. The lower limit of SPEED is drawn by $N = 1$. Increasing S_{T} reduces SPEED. The influence of N on SPEED is higher at low S_{T}. The observed trend of SPEED is similar to the observed trend in Figure 4.1. An increase of N results in additional increase of SPEED which is more

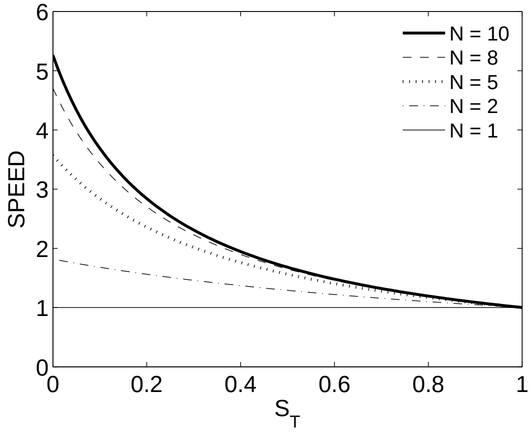

Figure 4.3: SPEED curves evolution with the used number of iterations, $\beta = 1.0$ and $\alpha_{\text{Iter}} = 0.1$.

significant at a lower value of N compared to the increase of SPEED at a higher value of N. To emphasize this issue, SPEED is plotted versus N in Figure 4.4. SPEED increases with a steep slope that drastically decreases with increasing N. The upper limit of SPEED, the horizontal dashed line, is found by substituting $S_{\text{T}} = 0.1$, $\alpha_{\text{Iter}} = 0.1$ and $\beta = 1.0$ in (4.23) yielding a value of 5.26.

The simplified upper limit, defined in (4.23), depends on α_{Iter}, β and S_{T}. In Figure 4.5, the upper limit curves are grouped between $\alpha_{\text{Iter}} = 1$ (lower) and $\alpha_{\text{Iter}} = 0$ (upper). The upper limit of $\alpha_{\text{Iter}} = 0$ goes to infinity at $S_{\text{T}} = 0$ as result of dividing by zero in (4.23). Again, the upper limit of SPEED decreases by increasing S_{T}. At the same S_{T}, the upper limit is shifted up by decreasing α_{Iter}. Increasing α_{Iter} has more significant influence in shifting the upper limit down at low S_{T} than at high S_{T}

The influence of elastic/plastic BUILD ratio β on the simplified SPEED upper limit is plotted in Figure 4.6. The curves are grouped between $\beta = 0$ (lower limit) and $\beta = 1$ (upper limit). A decrease of the curves is observed with increasing S_{T}. β has larger influence on the curves at lower value of S_{T} compared to high value of S_{T}, which is typical behavior.

4.3 Three domain analytical formula

The three domain efficient implicit method has an advantage over the two domain efficient implicit method. This advantage is adding an additional update frequency strategy, the multi-incremental strategy, to the iterative and the incremental update strategy. The multi-incremental strategy has a similar update procedure as the incremental update strategy except it is extended over a number of increments (Section 2.3.3). The cost of the three domain method for the number of iterations that covers a number of increments can be split into four parts. These parts are the cost of solving the entire system of equations for all

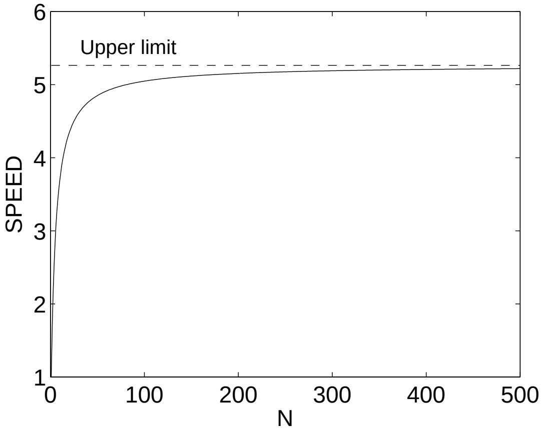

Figure 4.4: SPEED curve evolution with the used number of iterations, $\beta = 1.0$, $S_T = 0.1$ and $\alpha_{\text{Iter}} = 0.1$, the upper limit is found at $N \to \infty$.

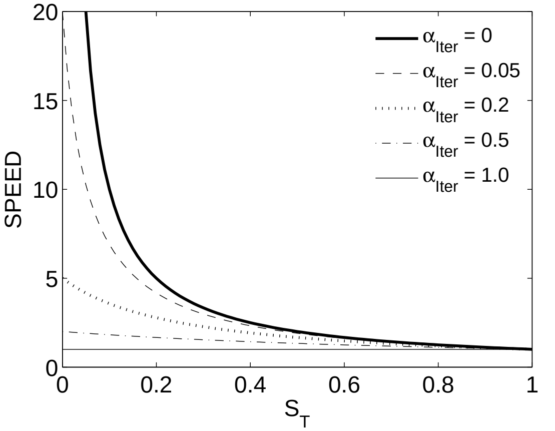

Figure 4.5: Simplified SPEED upper limit curves evolution for different iterative ratio at $\beta = 1.0$ and $N \to \infty$.

iterations, BUILD and UPDATE of the iterative domain (super elements) for all iterations, BUILD and UPDATE of the incremental domain that is performed every increment and the BUILD and UPDATE of the multi-incremental domain that is performed once for a number

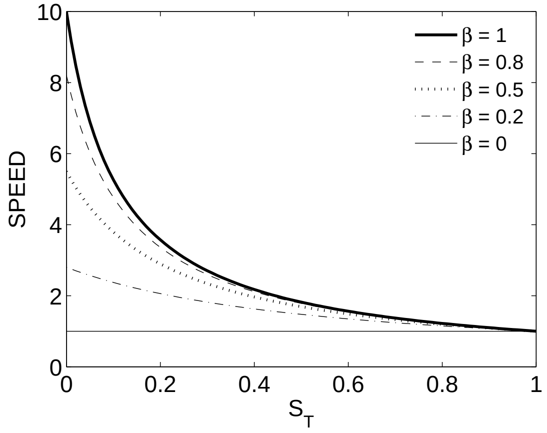

Figure 4.6: Simplified SPEED upper limit curves evolution for different elastic/plastic BUILD ratio β at $\alpha_{\mathrm{Iter}} = 0.1$ and $N \to \infty$.

of increments.

The speed factor (SPEED) is defined as the ratio of the cost for the standard Newton method to the cost of the three domain method for the same number of iterations

$$\mathrm{SPEED} = \frac{N}{NS_{\mathrm{T}} + \mathrm{CITER_T} + \mathrm{CINCR_T} + \mathrm{CMINCR_T}} \tag{4.24}$$

where

$$\mathrm{CITER_T} = \left(\frac{B_{\mathrm{T}}}{\alpha_{\mathrm{P}} + (1-\alpha_{\mathrm{P}})\beta} + \frac{U_{\mathrm{T}}}{\alpha_{\mathrm{P}} + (1-\alpha_{\mathrm{P}})\eta}\right) N\alpha_{\mathrm{Iter}} \tag{4.25}$$

$$\mathrm{CINCR_T} = \left(\frac{\beta B_{\mathrm{T}}}{\beta\alpha_{\mathrm{E}} - \alpha_{\mathrm{E}} + 1} + \frac{\eta U_{\mathrm{T}}}{\eta\alpha_{\mathrm{E}} - \alpha_{\mathrm{E}} + 1}\right) \frac{N\,\alpha_{\mathrm{Incr}}}{N_{\mathrm{Incr}}} \tag{4.26}$$

$$\mathrm{CMINCR_T} = \left(\frac{\beta B_{\mathrm{T}}}{\beta\alpha_{\mathrm{E}} - \alpha_{\mathrm{E}} + 1} + \frac{\eta U_{\mathrm{T}}}{\eta\alpha_{\mathrm{E}} - \alpha_{\mathrm{E}} + 1}\right) \frac{N\,\alpha_{\mathrm{MIncr}}}{N_{\mathrm{MIncr}}} \tag{4.27}$$

where N is the number of iterations that spans more than one load increment. α_{Mincr} defines the ratio of the multi-incremental super elements to the total super elements in the FE model. The update is performed for the incremental and the multi-incremental super elements after N_{Incr} and N_{MIncr} iterations, respectively. It is assumed in (4.24) that the variation of the Newton iteration cost is negligible. In (4.25), the iterative ratio α_{Iter} can substitute the plastic ratio α_{P}. The elastic ratio α_{E} in (4.26) and (4.27) consists of both the incremental and multi-incremental ratio $\alpha_{\mathrm{E}} = \alpha_{\mathrm{Incr}} + \alpha_{\mathrm{MIncr}}$. It is conditionally required that N_{Incr} and N_{MIncr} are smaller or equal to N and N_{Incr} is smaller or equal to N_{MIncr}

For simplicity, the elastic/plastic BUILD ratio β and the elastic/plastic UPDATE ratio η can be assumed equal. This simplifies $\mathrm{CITER_T}$, $\mathrm{CINCR_T}$ and $\mathrm{CMINCR_T}$ and consequently the analytical formula of SPEED for the three domain method

$$\mathrm{CITER_T} = \Big(\frac{1 - S_\mathrm{T}}{\alpha_\mathrm{Iter} + (1 - \alpha_\mathrm{Iter})\beta}\Big) N\alpha_\mathrm{Iter} \tag{4.28}$$

$$\mathrm{CINCR_T} = \Big(\frac{(1 - S_\mathrm{T})\beta}{\beta\alpha_\mathrm{E} - \alpha_\mathrm{E} + 1}\Big) \frac{N\,\alpha_\mathrm{Incr}}{N_\mathrm{Incr}} \tag{4.29}$$

$$\mathrm{CMINCR_T} = \Big(\frac{(1 - S_\mathrm{T})\beta}{\beta\alpha_\mathrm{E} - \alpha_\mathrm{E} + 1}\Big) \frac{N\,\alpha_\mathrm{MIncr}}{N_\mathrm{MIncr}} \tag{4.30}$$

The analytical formula of the two domain method is a special case of the three domain method. The three domain model is reduced to the two domain model if it uses no multi-incremental super elements or it uses the same number of iterations to update the incremental and multi-incremental super elements. Substituting $\alpha_\mathrm{MIncr} = 0$ in (4.27) results in $\mathrm{CMINCR_T} = 0$. The elastic ratio becomes $\alpha_\mathrm{E} = \alpha_\mathrm{Incr} = 1 - \alpha_\mathrm{Iter}$ that reduces SPEED model (4.24) to the two domain SPEED model that is presented in (4.13). The second case is applying the same number of iterations N_E for updating the incremental and multi-incremental super elements. This is equivalent to (4.13) with the use of $\alpha_\mathrm{E} = \alpha_\mathrm{Incr}$. The same implies to the simplified three domain SPEED.

Another special case is an elastic domain made entirely of multi-incremental or incremental updated super elements. The incremental super element requires more update compared to the multi-incremental super elements and that results in a higher SPEED curve for $\alpha_\mathrm{E} = \alpha_\mathrm{MIncr}$ than $\alpha_\mathrm{E} = \alpha_\mathrm{Incr}$, Figure 4.7. The demonstration is carried with $\alpha_\mathrm{Iter} = \alpha_\mathrm{P} = 0.25$, $N = N_\mathrm{MIncr} = 20$, $N_\mathrm{Incr} = 5$ and $\beta = \eta = 1$. Four UPDATE steps are performed in case of $\alpha_\mathrm{E} = \alpha_\mathrm{Incr}$ while only one is performed in case of $\alpha_\mathrm{E} = \alpha_\mathrm{MIncr}$. The achieved SPEED curve for $\alpha_\mathrm{E} = \alpha_\mathrm{MIncr}$ is the same like the SPEED curve of two domain method for $N = 20$. The advantage of less update is more beneficial at low $\mathrm{S_T}$.

In the following case, the influence of N_MIncr on SPEED is investigated on a FE model that has all types of update strategies. The increase of N_MIncr reduces the number of updates within a number of iterations N/N_MIncr. This reduces the contribution of $\mathrm{CMINCR_T}$ in the three domain method in the denominator in (4.24), resulting in an increase of SPEED. SPEED curves are plotted in Figure 4.8 for a model with the following settings: $\alpha_\mathrm{Iter} = \alpha_\mathrm{P} = 0.1$, $\alpha_\mathrm{Incr} = \alpha_\mathrm{MIncr} = 0.45$, $N = 100$, $N_\mathrm{Incr} = 5$ and $\beta = \eta = 1$. Several values of N_MIncr are used $5, 10, 20, 100$ iterations that correspond to $20, 10, 5, 1$ updates of the multi-incremental super elements. The increase of N_MIncr shifts up the SPEED curve. The lower SPEED curve represents the two domain SPEED curve since a similar number of iterations are used for both the incremental and the multi-incremental super elements. The upper SPEED curve is a result of updating the super elements only once for 100 iterations. As expected, the performance of the SPEED curves is better at low $\mathrm{S_T}$ and it decreases with increasing $\mathrm{S_T}$ until it reaches one at $\mathrm{S_T} = 1$.

4.4 Case study

In this section, the analytically derived SPEED curves are validated for a two domain method. The simulation of one loop of a SPIF process, introduced in Section 3.3.1, is used for the validation. The numerical model is made of 6400 shell elements based on discrete Kirchhoff

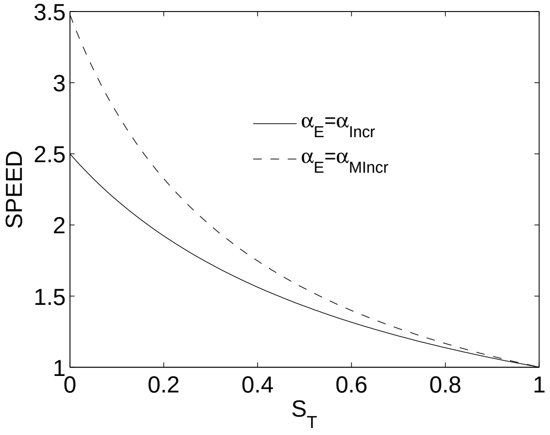

Figure 4.7: SPEED curves evolution for two special cases of elastic ratio α_E. $\alpha_{Iter} = \alpha_P = 0.25$, $N = N_{MIncr} = 20$, $N_{Incr} = 5$ and $\beta = \eta = 1$

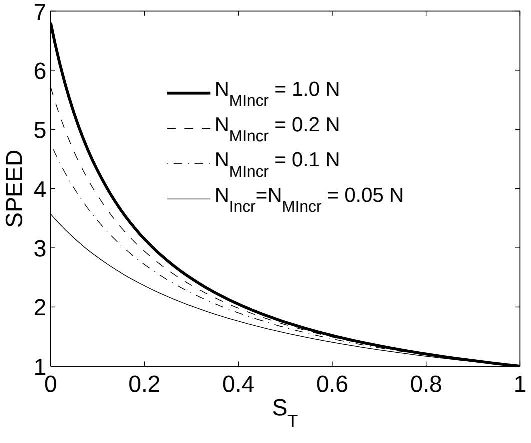

Figure 4.8: SPEED curves evolution for different N_{MIncr}. $\alpha_{Iter} = \alpha_P = 0.1$, $N = 100$, $N_{Incr} = 5$ and $\beta = \eta = 1$, $\alpha_{Incr} = 0.45$, $\alpha_{MIncr} = 0.45$

triangle element. The material model is introduced in Section 2.2.1. The numerical data is extracted out of the standard Newton implicit simulation. During the simulation, the cost of the last iteration for each increment is recorded. The iteration cost and the corresponding

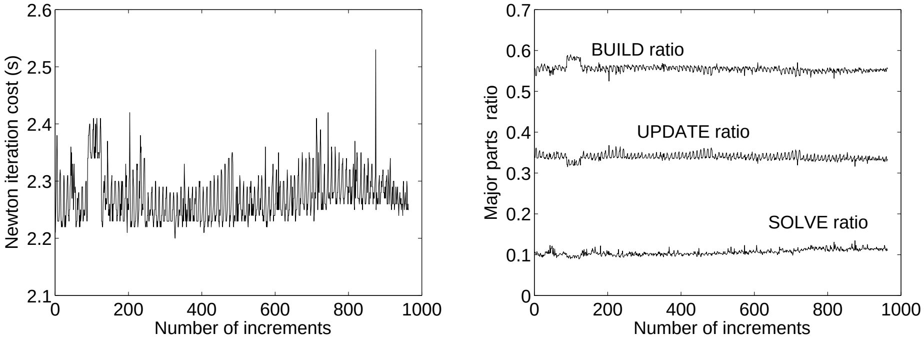

Figure 4.9: The last iteration cost of each increment (left) and the major parts ratio (right). The data is related to the one loop standard simulation for SPIF process.

Table 4.2: CPU time for BUILD and UPDATE.

	Full plastic	Full elastic	SPIF	
BUILD (s)	1.56	1.18	1.26	$\beta = 0.76$
UPDATE (s)	2.24	0.73	0.77	$\eta = 0.33$

major parts ratio (B_T, U_T and S_T) are plotted in Figure 4.9. The iteration costs on average 2.27 s. the averaged major parts ratio B_T, U_T and S_T are 0.555, 0.339 and 0.106 respectively. An unexpected slight increase of BUILD cost is observed for increments 89−127.

In order to define the elastic/plastic BUILD ratio β and elastic/plastic UPDATE ratio η, a stretching test is performed on the same FE model. The stretching mechanism introduces a global uniform deformation in the blank (FE model) that insures fully plastic deformation for all integration points in the FE model. The required CPU time to perform B^P and U^P for fully plastic deformation are 1.56 s and 2.24 s. The cost of a completely elastic deformation B^E and U^E are 1.18 s and 0.77 s. the elastic/plastic BUILD ratio β and elastic/plastic UPDATE ratio η are calculated by (4.4) and (4.7) that results in $\beta = 0.76$ and $\eta = 0.33$, the related data is summarized in Table 4.2. In SPIF simulation, the cost of BUILD and UPDATE are 7 % and 5 % larger than the corresponding elastic cost. This indicates that only a small part of the model is deforming plastically.

The number of iterations per increment is also recorded during the standard Newton simulation, Figure 4.10. Almost half of the increments requires 3 iterations while the second half requires 2 iterations. Some increments require at the initial stage and when the tool approaches the corners 5−6 iterations. The plastic and the elastic ratio are assumed to be equal to the iterative and the incremental ratio that are observed in the Iter–Incr super

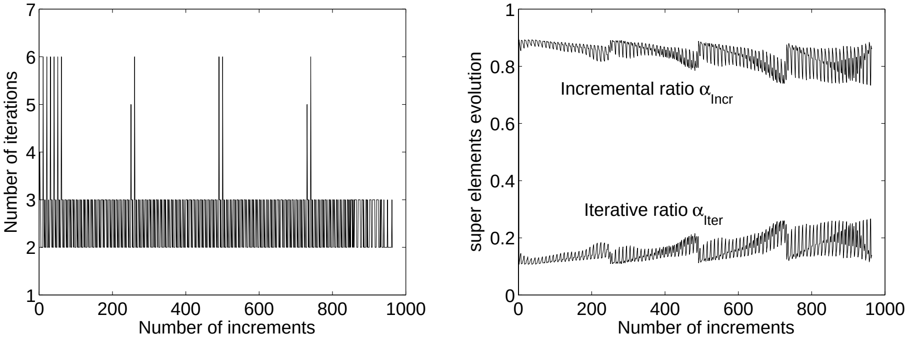

Figure 4.10: The iteration evolution of the standard simulation (left) and the super element classification for the Iter–Incr simulation (right).

element based simulation of the one loop SPIF process. The evolution of the iterative and the incremental ratio is explained in Section 3.3.1 and plotted again here for convenience in Figure 4.10.

Now, all the variations of the two domain analytical formula SPEED are available. The variations are substituted in (4.13) and the expected analytical SPEED is plotted in Figure 4.11. Despite the observation of almost constant major parts ratio, the analytical predicted SPEED is oscillatory. Some peaks are observed in SPEED that coincides with the large number of iterations for those increments. For the rest of the increments, the range of SPEED variations corresponds to $2-3$ iterations. SPEED is inversely proportional to the iterative ratio, the increase of α_{Iter} results in a decrease in SPEED and vice versa. The averaged analytical SPEED is 1.64. The practically observed SPEED is 1.68 that is calculated by dividing the overall CPU time of the standard Newton implicit simulation over the super element based simulation CPU time. The analytical formula of SPEED has a very good agreement with the practically observed SPEED. This validates the model and it can predict in advance the expected acceleration of an implicit simulation by the implementation of the super element based efficient implicit approach.

The cost of BUILD and UPDATE is influenced by the element type. This has consequently an effect on the speeding factor. To demonstrate this, the one loop SPIF simulation is performed again but with shell elements based on discrete shear triangle DST (Batoz and Lardeur, 1989) instead of discrete Kirchhoff triangle DKT (Batoz *et al.*, 1980). Both elements have 6 degrees of freedom per node. The DST element implicitly couples the lateral displacement with the rotation of a midsurface-normal line by transverse shear stiffness of the material, while in the DKT element it is explicitly coupled by enforcing zero transverse shear strain at selected locations (Cook *et al.*, 2002).

A slightly larger iterative ratio is observed in the simulation based on the DST element. This indicates that almost the same speeding factor (1.68) is expected but a higher speeding factor of 1.89 is achieved. The computing time of the simulations is listed in Table 4.3.

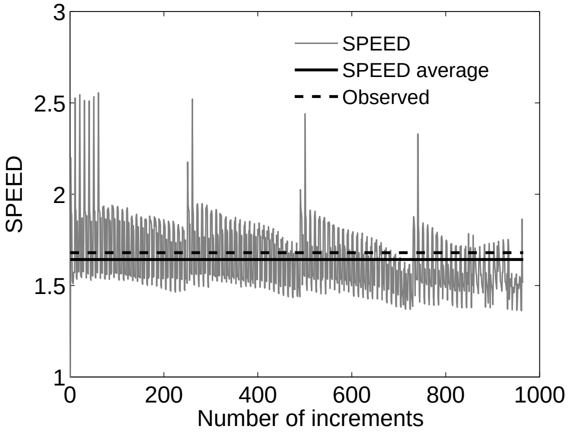

Figure 4.11: The two domain analytical SPEED.

Table 4.3: The overall computing time and the corresponding speeding factor for one loop of SPIF simulation.

	Newton $_{\mathrm{DKT}}$	Iter–Incr $_{\mathrm{DKT}}$	Newton $_{\mathrm{DST}}$	Iter–Incr $_{\mathrm{DST}}$
CPU time (s)	5693.7	3393.2	8425.4	4456.8
Speeding factor	1.00	1.68	1.00	1.89

Almost the same number of iterations is required by both simulations but the DST based simulation requires more CPU time to complete the simulation compared to the DKT simulation CPU time. In DST based simulation, BUILD and UPDATE require 1.6 and 1.33 s while they require 1.13 and 0.74 s in the DKT based simulation, respectively. A comparison of the major iteration cost for both simulations is shown in Figure 4.12. Solving the system of equations requires the same CPU time in both simulations. Considering the cost ratio perspective, an increase is observed in UPDATE cost ratio for the DST simulation compared to the DKT ratio while a decrease is observed in BUILD and SOLVE. This results in an overall increase of the achieved SPEED for the DST based simulation compared to the achieved SPEED for the DKT based simulation

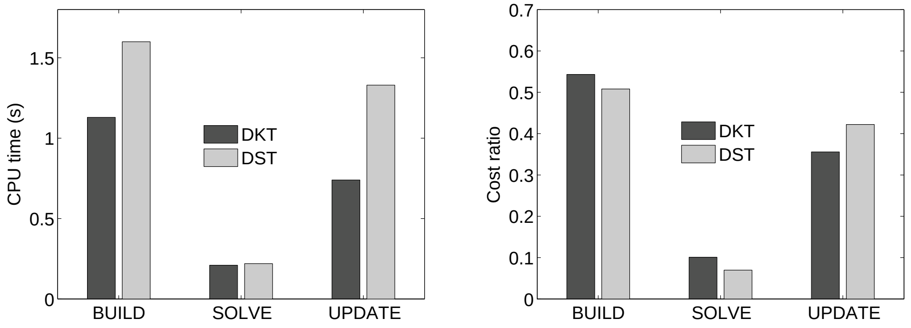

Figure 4.12: A comparison of the major iteration cost (left) and cost ratio (right) between the DKT based simulation and the DST based simulation. The results are presented for the first iteration of increment number 100 in the standard Newton simulation.

4.5 Summary and conclusions

The speed factor (SPEED) measures the efficiency of the super element based efficient implicit approach in accelerating the standard implicit simulation of localised deformation processes. SPEED is defined as the CPU time cost of one Newton increment to the cost of one increment of the efficient implicit time integration procedure. SPEED is influenced by several factors. These factors are the number of performed iterations, the combination of the different update strategy ratios, the used update strategies and the major parts cost of the Newton iteration (BUILD, UPDATE and SOLVE). The major parts ratio of Newton iteration depends on the material model and the element type.

An analytical formula that combines all of these factors is developed in order to understand the computational performance of the efficient implicit simulation for localised deformation. In Section 4.2, The two domain analytical formula of SPEED is developed. The incremental cost of the two domain efficient implicit method can be split into three parts: the cost of the iterative domain, the cost of solving the entire system of equations and the cost of the incremental domain. The cost of solving the system of equations and the iterative domain are linearly scaled by the number of iterations. The incremental domain cost is consumed once per increment. The iterative and the incremental domain costs are functions of the iterative ratio, BUILD and UPDATE ratio and the elastic/plastic ratios β and η. SPEED is enhanced by reducing either the SOLVE ratio or the iterative ratio. Also, SPEED performs better with a larger number of iterations or by increasing the elastic/plastic ratios. A simplified upper limit of SPEED is found to be inversely proportional to the SOLVE ratio and the iterative ratio. Theoretically, the SPEED can go to infinity at negligible SOLVE ratio combined with zero iterative ratio.

The analytical formula of SPEED is extended in order to predict the efficiency of the three domain approach (Section 4.3). Now, SPEED is defined as the ratio of cost for a number of increments with the standard Newton method to the cost of the three domain method for the same number of increments. The cost for the three domain method can be split into four parts: the cost for the iterative domain, the incremental domain, the multi-incremental domain and the cost of solving the entire system of equations. Solving the system of equations and updating the iterative domain are performed every iteration. The incremental domain is updated every increment while the multi-incremental domain is updated only once. The SPEED curve of the three domain method is higher than the curve of the two domain method, it accelerates the implicit method more than the two domain method for the same number of increments. The three domain algorithm performance has similar response as the two domain algorithm regarding the iterative ratio, the SOLVE ratio and the elastic/plastic ratio β.

A demonstrative case study is analyzed in Section 4.4. The two domain analytical formula of SPEED is applied to the implicit simulation of performing one loop of SPIF process. The two domain analytical formula of SPEED has a very good correspondence with the practically observed speed factor. The average predicted SPEED for the simulation is 1.64 while the two domain method practically accelerates the simulation by 1.68. This verifies that the model is robust and it can predict in advance the expected SPEED of an implicit simulation for localised deformation.

It becomes clear that the efficient implicit method is suitable for intermediate scale simulation. The SOLVE cost is a crucial factor regarding the achieved SPEED. For this reason, numerical techniques are applied in the following chapter in order to keep SOLVE ratio as low as possible during the implicit simulation. This will enhance the achieved speed factor.

5. Static condensation and remeshing

The performance of the efficient implicit method can be enhanced by coupling the method with other numerical techniques. In this chapter, two numerical techniques are emphasized. The first technique is the static condensation (Section 5.1). It reduces the size of the system of equations and that may accelerate the computation by reducing the time consumed by the solver. In Section 5.2, the h-adaptivity is investigated. It keeps the FE mesh as small and efficient as possible. One level of refining and coarsening is studied for the SPIF process simulation.

5.1 Static condensation

Static condensation is used to reduce the size of the system of equations. This application had a major advantage in structural engineering, specially for analyzing large structures that were often beyond the capacity of the computers in the past. Substructuring (super element) is a way to organize the static condensation of large linear systems arising from the discretization of partial differential equations (Smith *et al.*, 1996). Introducing the super element discretization of a FE model in Section 3.1 provides the background to statically condense the super elements. For the incrementally and the multi-incrementally updated strategies, the tangent stiffness matrix is constant for an increment or a number of increments. The internal force vector is linearly updated by the multiplication of the tangent stiffness matrix and the incremental displacement. This pseudo-linear treatment is applied after the nonlinear update of the geometrical and the material nonlinearities in the previous steps. The incremental and the multi-incremental super elements are suitable for static condensation because of the pseudo-linear treatment. The iterative update strategy forces a full nonlinear Newton update every iteration and that does not fit with the linearity assumption of static condensation.

The super element degrees of freedom can be classified into master DOFs and slave DOFs as shown in Figure 5.1. For pseudo-linear super elements, the master DOF is defined as the external DOF that may be connected to another super element and the slave DOF is an internal DOF that belongs only to one super element. All DOFS for the iterative treated super element are master DOF. The pseudo-linear super element tangent stiffness matrix, incremental displacement and internal force vector are assembled based on the master–slave classification as

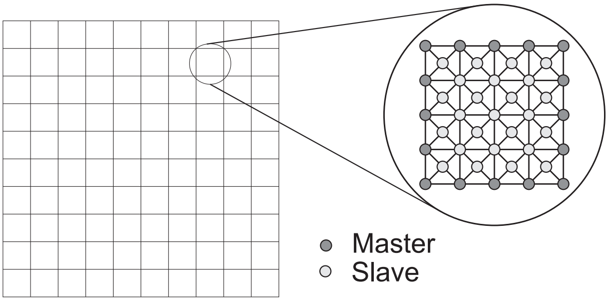

Figure 5.1: The classification of the super element DOFs into master and slave DOFs.

$$\underbrace{\begin{bmatrix} K_{ss} & K_{sm} \\ K_{ms} & K_{mm} \end{bmatrix}^{\text{Super}}}_{K_{\text{int}}^{\text{Super}}} \underbrace{\begin{pmatrix} d_s \\ d_m \end{pmatrix}^{\text{Super}}}_{d^{\text{Super}}} = \underbrace{\begin{pmatrix} f_s \\ f_m \end{pmatrix}^{\text{Super}}}_{f_{\text{int}}^{\text{Super}}} \tag{5.1}$$

where the subscript m and s refer to master and slave DOFs, respectively. The slave (internal) DOFs are statically condensed to their master (external) DOFs

$$d_s^{\text{Super}} = (K_{ss}^{\text{Super}})^{\text{-1}}(f_s^{\text{Super}} - K_{sm}^{\text{Super}}\, d_m^{\text{Super}}) \tag{5.2}$$

Such that the condensed form becomes

$$K_{\text{int}}^{\text{Super,c}}\, d_m^{\text{Super}} = f_{\text{int}}^{\text{Super,c}} \tag{5.3}$$

where

$$K_{\text{int}}^{\text{Super,c}} = K_{mm}^{\text{Super}} - K_{ms}^{\text{Super}}\,(K_{ss}^{\text{Super}})^{\text{-1}}\, K_{sm}^{\text{Super}} \tag{5.4}$$

$$f_{\text{int}}^{\text{Super,c}} = f_m^{\text{Super}} - K_{ms}^{\text{Super}}\,(K_{ss}^{\text{Super}})^{\text{-1}}\, f_s^{\text{Super}} \tag{5.5}$$

Now, the condensed global tangent stiffness matrix is assembled of the condensed pseudo-linear super elements and the iterative super elements tangent stiffness matrix

$$K_{\text{int}}^{\text{Glob,c}} = \sum_{i=1}^{IT} K_{\text{int,i}}^{\text{Super}} + \sum_{i=1}^{PL} K_{\text{int,i}}^{\text{Super,c}} \tag{5.6}$$

where IT and PL are the number of the iterative super elements and the pseudo-linear super elements, respectively. Similarly, the condensed global internal force vector is assembled.

$$f_{\text{int}}^{\text{Glob,c}} = \sum_{i=1}^{IT} f_{\text{int,i}}^{\text{Super}} + \sum_{i=1}^{PL} f_{\text{int,i}}^{\text{Super,c}} \tag{5.7}$$

The size of the global condensed system of equations is smaller than the original size of the global system of equations (non-condensed). This difference in size is a result of condensing the internal DOFs of the pseudo-linear super elements, both the incremental and multi-incremental super elements. On the other hand, the global condensed system of equations is denser than the original system of equations. Synchronized with the update strategy, the static condensation procedure is performed only once for the incremental super element and once for a number of increments for the multi-incremental super element. This is performed after the nonlinear update of the tangent stiffness matrix and the internal force vector. After the convergence of the load increment, the pseudo-linear super element slave DOFs are evaluated by (5.2).

The advantage of applying the static condensation is to reduce the time required for solving the system of equations. As discussed previously, the two domain method or the three domain method has no influence on the cost of solving the system of equations (SOLVE) because all DOFs are retained. Keeping in mind that solving the system of equations is performed every iteration and considering the two domain method that is introduced in Section 2.3.2 and the related analytical formula presented in Section 4.2, the condensation procedure becomes an additional cost to the incremental cost of the two domain method. The analytical formula is extended to include the condensation cost and the new cost of solving the condensed system of equations as

$$\mathrm{SPEED_c} = \frac{NT}{\underbrace{\mathrm{CON} + NS_\mathrm{c}}_{NS} + \mathrm{CITER} + \mathrm{CINCR}} \tag{5.8}$$

where CON is the cost of performing the condensation and S_c is the cost ratio of solving the condensed system of equations. The condensation cost and the cost of solving the condensed system of equations every iteration replace the cost of solving the original system of equations every iteration. The cost of building the system of equations and updating the stresses of the iterative and the incremental part of the FE model (CITER, CINCR) are independent of the used strategy to solve the system of equations, condensed or not. In order to benefit from solving a condensed system, the sum of condensation cost and the cost of solving the condensed system of equations has to be less than the cost of solving the original system of equations

$$\mathrm{CON} + NS_\mathrm{c} < NS \tag{5.9}$$

This results in a smaller overall incremental cost (the denominator in (5.8)), and that increases the achieved SPEED. Otherwise, if the resulting cost of the condensation and solving the condensed system is larger than solving the original system, SPEED is reduced compared to the non-condensed solution method. This can be a result of the large cost of the condensation procedure or an increase in the cost of solving the condensed system of equations because it is less sparse (denser).

Similarly, the three domain analytical formula of SPEED can be extended to include the influence of the condensation. It becomes

$$\mathrm{SPEED} = \frac{NT}{\underbrace{\frac{N\,\mathrm{CON_{Incr}}}{N_\mathrm{Incr}} + \frac{N\,\mathrm{CON_{MIncr}}}{N_\mathrm{MIncr}} + NS_\mathrm{c}}_{NS} + \mathrm{CITER} + \mathrm{CINCR} + \mathrm{CMINCR}} \tag{5.10}$$

where CON_{Incr} and $\text{CON}_{\text{MIncr}}$ are the cost of condensing the incremental super elements and the multi-incremental super elements, respectively. Both are scaled with the same ratio as CINCR and CMINCR, (4.26) and (4.27) respectively. The multi-incremental update strategy has lower update frequency compared to the incremental update strategy and definitely the condensation is performed in a lower frequency for multi-incremental super elements compared to the incrementally updated super elements.

The condensation is performed based on the LU-factorization method (Kreyszig, 1993). It is a relatively less expensive method to solve a system of equations compared to the standard Gauss elimination. The system matrix (A) is decomposed into an upper triangular matrix (U) and a lower triangular matrix with diagonal of ones (L) as

$$Ax = LUx = Ly = b \tag{5.11}$$

where x is the unknown vector solved from $Ux = y$ and b is the right-hand side. Using the LU-factorization method to derive $(K_{ss}^{\text{Super}})^{\text{-1}} K_{sm}^{\text{Super}}$ is more efficient than determining $(K_{ss}^{\text{Super}})^{\text{-1}}$ explicitly if the number of the slaves is larger than the number of masters. Because K_{ss}^{Super} is a symmetric positive definite matrix, it is found that $U = L^{\text{T}}$ without imposing conditions on the main diagonal. This special case of the LU-factorization method is known as Cholesky's method.

Actually, f_s^{Super} and each column of K_{sm}^{Super} can be used as the right-hand side vector b. By applying this, the $(K_{ss}^{\text{Super}})^{\text{-1}}\ K_{sm}^{\text{Super}}$ and $(K_{ss}^{\text{Super}})^{\text{-1}}\ f_s^{\text{Super}}$ are found. These operations are performed using Sun Performance Library$^{\text{TM}}$ package that is developed for sparse linear systems of equations.

5.1.1 Case study

In this case study, an implicit simulation of drawing a line in a blank by a SPIF process is simulated. An initially flat numerical blank of 300 × 300 × 1.2 mm^3 is discretized by 14400 discrete shear triangular shell elements. Through the thickness of the element 5 integration points are used (in total 15 per element). The drawn line is 1 mm deep and 260 mm long. It is parallel to the blank edge and 20 mm away from the edge. The blank edges are clamped. The line is drawn by an analytical spherical tool with a 20 mm diameter. The focus of this case study is to investigate the computational performance of the condensation procedure. For that, a reference simulation is performed by the standard Newton method. The two domain method is used as efficient approach to accelerate the standard simulation. The condensed two domain approach (Condensed) performance is compared to the non-condensed two domain method (Original).

The standard Newton implicit simulation performs 352 load increments and it requires on average 3 iterations per increment to converge and 168 line searches are performed in total. The overall CPU time for this simulation is 6574 s (1.82 hrs). The averaged cost for a Newton iteration is 5.69 s that is split into 49.6 %, 12 % and 38.4 % for building the system of equations, solving it and updating the stresses respectively. For the two domain method, 4 super elements are classified for the iterative update strategy and the rest of the super elements are classified for the incremental update strategy. Different sizes of the super element are considered that are 4, 16, 36, 64, 100, 144 and 400 elements per super element. They are referred as S4, S16, S36, S64, S100, S144 and S400, respectively. This corresponds to

Table 5.1: The performance of the original and the condensed two domain simulations.

	α_{Iter}	Condensed CPU (s)	Original CPU (s)	Condensed SPEED	Original SPEED
S4	0.001	2530	2499	2.55	2.57
S16	0.004	2561	2463	2.51	**2.61**
S36	0.010	2733	2545	2.36	2.53
S64	0.018	2943	2583	2.19	2.49
S100	0.028	3296	2625	1.95	2.45
S144	0.040	3824	2677	1.68	2.41
S400	0.111	8481	2971	**0.76**	2.16

group the FE mesh into 3600, 900, 400, 225, 144 and 36 super elements. Constraining the iterative domain to include 4 super elements influences the achieved accuracy, but in this case study that is not of interest here.

The main focus is to investigate the achieved SPEED for the Condensed and the Original two domain method. The achieved SPEED for different super elements is summarized in Table 5.1. As expected, the increase of the super element size results in a decrease of the achieved SPEED for both two domain methods because of the corresponding increase in the iterative ratio (always 4 super elements are iteratively treated). An unexpected increase of SPEED is observed in the Original method at S16. In general, the Original two domain method has a higher achieved SPEED than the Condensed two domain method except at S4 where they are almost equal. The best achieved SPEED is 2.57 for S4 by the Original two domain method. The lowest performance of the Original method SPEED is observed for S400 and it is 2.16. In the Condensed two domain method, the increase of the super element size results in more significant decrease of the achieved SPEED. At S400, the Condensed two domain SPEED is 0.76 (less than one) that violates the golden rule of the efficient implicit simulation method. At this setting, the condensed two domain method slows down the implicit simulation instead of accelerating it.

The main difference between the presented two domain methods is performing the static condensation or not. As explained, the SPEED performance is influenced by the condensation cost and the cost of solving the condensed system of equations. The cost of condensation and the cost of solving the condensed system of equations for different super elements are shown in Figure 5.2. For S400, the static condensation results in an increase of solver cost almost by 300 %. The condensation cost is 5 times the cost of solving the original system of equations (strange!). This explains the bad achieved SPEED (0.76), slowing the standard Newton implicit simulation. In case of S144 and S100, the cost of solving the condensed system of equations shows a slight increase or is almost equal to the cost of solving the original system of equations. Because of the cost of the condensation of these super element sizes, the achieved SPEED for the Condensed method is going to be

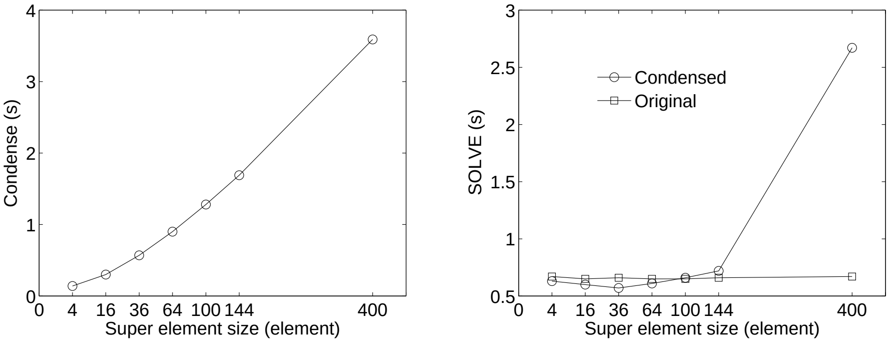

Figure 5.2: The cost of condensing (left) and solving (right) the system of equations for different super element sizes.

less than the original SPEED.

A reduction in the cost of solving the condensed system of equations is observed for S4-100. A better SPEED can be expected for these settings. Now, the condensation cost determines the increase or the decrease of the achieved SPEED compared to the Original SPEED. Checking (5.9), the sum of the condensation cost and solving of the condensed system of equations for all settings is larger than the time required to solve the original system of equations for 3 iterations, as shown in Figure 5.3. To conclude: the static condensation is efficient only if the cost of solving the condensed set of equations is significantly reduced and it has to compensate for the cost of the condensation. The use of condensation has to be investigated to avoid a lower achieved SPEED. The presented description of the static condensation is deficient.

5.2 Adaptive remeshing

The crucial issue in single point incremental forming simulation is performing thousands of load increments on a relatively fine FE mesh. The small radius of the forming tool requires a fine mesh for the small contact area but is not required for the entire workpiece. The small contact area travels all over the workpiece in order to introduce the incremental deformation (it is a special feature of SPIF process). Basically, the FE model is made fine enough to capture the introduced deformation despite the knowledge of the current location of the forming tool. This is an inefficient modelling description of the workpiece. Alternatively, a relatively coarse FE model for the entire model combined with small traveling fine mesh would be more efficient.

The h-adaptivity method fits the efficient modelling needs. It is based on adapting the number of grid points and changing the mesh connectivity (Huerta *et al.*, 1998). Grid points are added to areas where more accuracy is demanded and it can be deleted in areas where the solution is accurate enough. In SPIF simulations, the h-adaptivity method is used to add

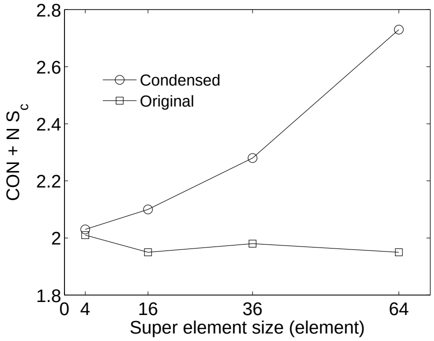

Figure 5.3: The cost of condensing and solving the system of equations for different super element sizes. For the Original method CON $= 0$ and $S_\mathrm{c} = S$.

grid points in the small area of contact. As the tool moves, more grid points are added to the FE model in the vicinity of the tool. The grid points added in the former tool location are deleted. This simulates the movement of a fine mesh embedded in a relatively coarse FE model.

Within this brief investigation, one level of h-adaptivity is implemented in the in-house FE package DiekA. Actually, the h-adaptivity refinement procedure of triangular shell elements by Meinders (2000) is extended by one level of coarsening. Depending on a particular remeshing criterion, a group of elements is nominated for refinement. The neighboring elements of these nominated group of elements have high potential to be refined in the following load increments. Both the nominated elements and their neighboring are called mother elements and refined once. Each mother element is divided into four equal elements, the newly created elements are called refined elements. To preserve mesh compatibility, each neighbor element of the mother elements is split into two equal elements. Any two split elements born out of the same mother element can be united and refined into four refined elements for the next load increments, if it is required. Meanwhile, the coarsening algorithm performs only on the refined and split elements. It degenerates the previous connectivity in a reverse order. This limits the growth of number of elements between the initial number of elements (lower limit) and refinement of the entire FE model once (upper limit). An example for element generation and degeneration is shown in Figure 5.4.

The refinement is performed when a specified error exceeds a threshold. An error indicator is used instead of an error estimator since it is computationally cheaper (Huerta *et al.*, 1998). Particularly, the geometrical error indicator developed by Bonet (1994) is used. It measures the variation of the geometry within the blank. A set of tangent axes is determined for each element and it is constant within the element. The variation of these sets of tangents from one element to its neighboring elements indicates the variation of the geometry which cannot be represented by the facet element. Therefore, a nodal averaging technique is used to quantify this variation. If the variation within a group of

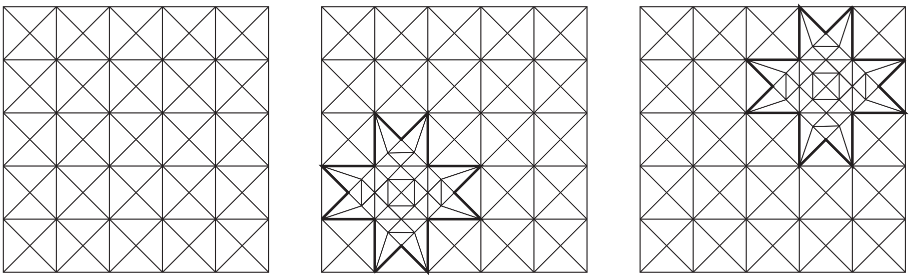

Figure 5.4: The description of the remeshing algorithm. The initial mesh (left) that is partially refined (middle). The coarsening of lower left part and the refinement of the upper right of the FE model (right).

elements exceeds a threshold, that group of elements is refined. On the other hand, if the variation within a group of refined elements is decreased, coarsening takes place on that group regenerating the mother elements.

Mapping

When a new grid connectivity is created, the state variables have to transfer from the old mesh to the new mesh. There are several approaches. One approach is using the old nodal value to evaluate the new nodal value, then determining the state variables at the integration points (Meinders, 2000). Another approach is to use a patch recovery, that depends on selecting specific locations within a group of elements, creating a smoothed field out of it and evaluate the new data. For instance, a bi-linear smoothed field requires at least four points to be selected within the group of elements. Further details on patch recovery using plane elements can be found in (Cook *et al.*, 2002). The chosen approach applies a least square approximation. The method fits a linear field based on all available integration points within a group of elements, not on selected points only. The location of the integration points and their values are used to create the linear field, then the location of the new integration point is used to determine its value.

Regardless of the used approach for mapping, data transfer predicts the exact value for the state variables when it maps mother element into split elements or refined elements. However, it either overestimates or underestimates the new value for the following remeshing cases: split into refined, refined into split, refined into mother and split into mother. The error is introduced because of fitting piecewise linear fields into one linear field. The least square method is the optimum approach, in the sense of accuracy. It minimizes the error during data transfer because of the use of all available integration points and it is not computationally expensive compared to the other approaches.

The correctness of implementing the least square method for mapping on one level of remeshing has to be validated before simulating the SPIF process. For that purpose, an initial linear field for the equivalent plastic strain is prescribed for a FE strip model. The

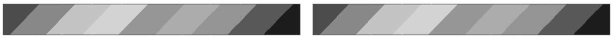

Figure 5.5: Linear field validation test for one level remshing using least square method.

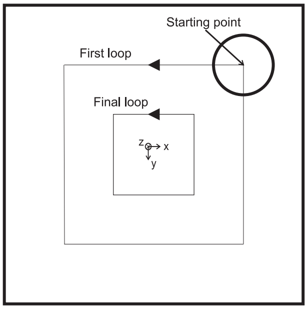

Figure 5.6: Tool path description of pyramidal shape SPIF process.

strip experiences 600 remeshing combinations. Theoretically and numerically within the machine accuracy, the linear field must remain the same. This is achieved in the one level of remeshing implementation test as shown in Figure 5.5.

5.2.1 Case study

A single point incremental forming process of a 45° pyramidal shape is simulated to verify whether the one level remeshing technique leads to significant CPU time reduction. The 17 mm deep pyramid is made out of a $100 \times 100 \times 1.2\,\text{mm}^3$ initially flat blank. An analytical spherical tool of 10 mm radius is used. The tool follows a counter-clockwise tool path for 34 loops. In each loop, the tool moves 0.5 mm vertically downwards. At a fixed vertical position, the tool performs the in-plane tool path. The simulation finishes when the tool reaches the end of loop 34, the first and the final loop description is shown in Figure 5.6.

The numerical blank is discretized with 3200 discrete Kirchhoff triangular shell elements. It is used as a reference model. Each element has in total 15 integration points. A simple material model is used (Section 2.2.1), representative for a mild steel. Two implicit simulations are performed using the h-adaptivity method. An intermediate coarse initial mesh of 800 triangular shell elements is used for these simulations. In the first simulation, the h-adaptivity introduces one level of refinement only (Refine) while one level of remeshing, refining and coarsening, is introduced in the second simulation (Remesh). During the h-adaptivity simulations, the number of elements is expected to vary between 800 and 3200.

The simulations are performed on a single core of DUAL Intel Xeon 3.06 GHz computer, the performance of the simulations are summarized in Table 5.2. The reference simulation,

Table 5.2: The simulations performance.

	Reference	Refine	Remesh
Nr. of increments	8087	8169	8154
Nr. of elements	3200	800-2032	800-2014
Nr. of Nodes	1681	441-1057	441-1048
CPU time (hr)	19.9	10.24	9.93
SPEED	1.0	1.94	2.00

performed on a fixed fine FE mesh, requires 19.9 hr to finish 8087 load increments while the remeshing simulation finishes 8154 load increments in 9.93 hr. The refinement only simulation performs 15 load increments more than the remeshing simulation and it requires 10.24 hr. The same terminology of SPEED in measuring the gained benefit in CPU time is used, dividing the reference CPU time cost over the used approach CPU time cost. The h-adaptivity approaches accelerate the reference simulation by almost the same factor of 2. The advantage of the h-adaptivity is to keep the number of grid points (nodes) and consequently the number of elements as low as possible.

The evolution of the grid points during the simulations are shown in Figure 5.7. The use of a geometrical variation indicator triggers the overall increase of number of nodes because of the shape development of the blank. By the use of the refinement approach only, an increase is observed in the number of nodes during intermediate stages of the simulation. The remeshing approach results in an increase of the grid points that is in general less than the growth of the number of nodes in the refinement approach. At the final stage of the process, the remeshing approach requires slightly less grid points than the refinement approach. The remeshing approach discretized the final geometry using 1048 nodes (2014 elements) while the geometry is discretized by 1057 nodes (2032 elements) using the refinement approach only. The final FE mesh of the simulations is shown in Figure 5.8.

In general, the results achieved by the refinement approach have a better agreement with the result achieved by the reference simulation compared to the results achieved by the remeshing approach. Considering the stretching strain at mid-integration point through thickness, The maximum achieved equivalent plastic strain at the reference FE mesh is 0.438. The refinement approach predicts almost the same maximum equivalent plastic (0.44) while it is overestimated by 6.4 % (0.466) using the remeshing approach (Figure 5.9). The maximum achieved equivalent plastic strain (reference) at the outer-integration point, that is representative of combined bending and stretching strain, is 0.875. It is overestimated slightly by 3 % (0.901) using the refinement approach and significantly by 19.1 % (1.042) using the remeshing approach. The better agreement of the refinement approach with standard simulation holds also for the achieved distribution of the equivalent plastic strain compared to the remeshing approach.

The remeshing approach performs 41 combinations of refining and coarsening during the simulation. Several elements have been refined and coarsened several times. For these

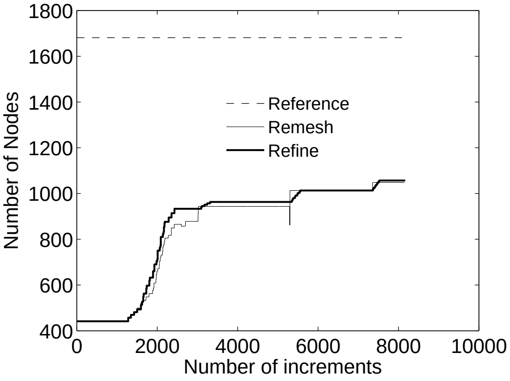

Figure 5.7: The evolution of the grid points for the pyramid SPIF simulation with different approaches.

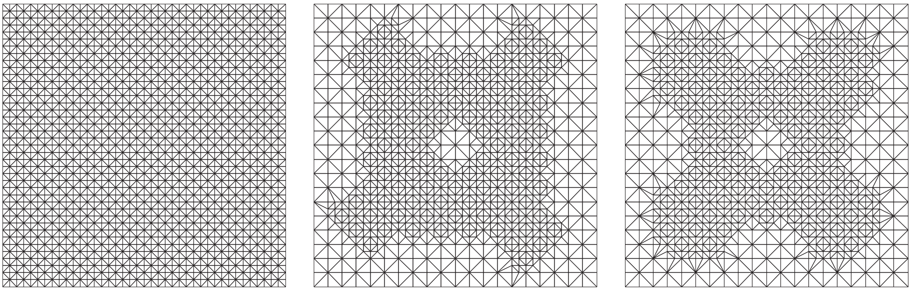

Figure 5.8: The reference FE mesh (left) and it is fixed for the entire simulation. The final stage FE mesh for the the refinement approach (middle) and the remeshing approach (right).

elements, applying the coarsening approach introduces errors because of smoothing the piecewise linear fields into one linear field. This error is larger at higher levels of strain than at lower levels of strains. This coarsening is followed by refinement in the following loop. Refinement results in significant deformation because of projecting the newly created grid points, those are in contact with the forming tool, to the tool surface to adapt to the tool geometry (Meinders, 2000). Mapping the data out of formerly smoothed fields results in further errors. This explains the significant overestimation at the outer integration point compared to the mid-integration point for the remeshing approach. To conclude: both h-adaptivity approaches successfully accelerate the reference simulation by a factor of 2. The results achieved by the refinement approach has a better prediction compared to the result

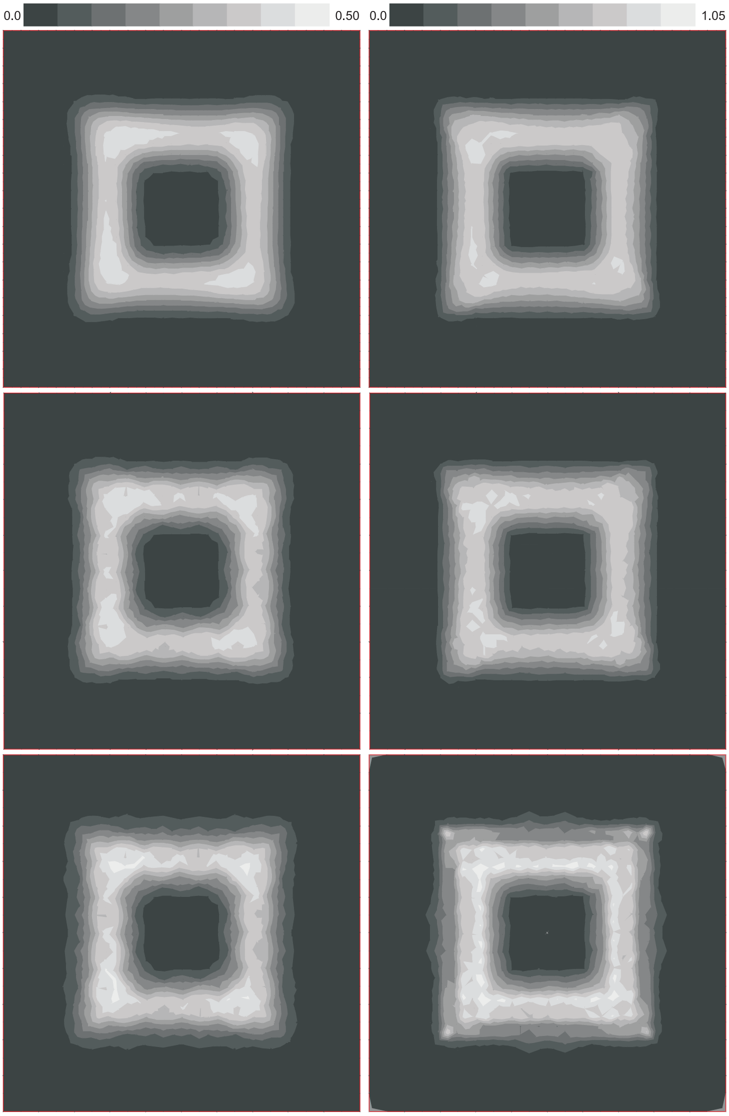

Figure 5.9: The achieved equivalent plastic strain at the mid-integration point through thickness (left) and the outer integration point (right). The results are achieved by the reference (top), refinement (mid) and remeshing approach (bottom).

achieved by the remeshing approach.

5.3 Summary and conclusions

The efficient implicit method applies a pseudo-linear treatment for the incremental and the multi-incremental super elements. Within this treatment for a number of increments, the tangent stiffness matrix is constant and the internal force vector is linearly updated using the fixed tangent stiffness matrix. Therefore, the static condensation of the internal degrees of freedom for pseudo-linear updated super elements is investigated. It reduces the global size of the system of equations and that is expected to reduce the CPU time used in solving the reduced system of equations. The static condensation is performed using an LU-factorization method. The implementation and the performance of the static condensation on the efficient implicit method is presented in Section 5.1. A demonstrative case study shows that the static condensation reduced the CPU time of solving the system of equations but not significantly to compensate the cost of condensing the system of equations. For some settings, the cost of solving the condensed system of equations becomes larger than the cost of solving the original system of equations. This results in reducing the achieved speeding factor by the efficient implicit method and even requiring more time than the standard implicit simulation.

The second numerical technique studied in this chapter is the h-adaptivity method, in Section 5.2. Particularly, one level of refining and coarsening is considered. Grid points are added to areas where more accuracy is demanded and they can be deleted in areas where the solution is accurate enough. The cost of simulating an intermediate coarse FE model is less than simulating the reference FE model and this is the main advantage. Repeated remeshing (refining and coarsening) for the SPIF process simulation results in overestimation of the equivalent plastic strain because of intensive mapping. Refining only maintains the accuracy compared to the reference FE model and it accelerates the reference simulation by the same factor as the remeshing method, it is two times faster than the reference model. The refinement approach has a high potential if it is integrated with the efficient implicit method to reduce the consumed time by the SPIF simulation by factors and maintaining the accuracy. A demonstrative case study will be presented in the next chapter investigating this claim.

6. Applications

Two real-life incremental forming processes are simulated for demonstration. The first application is to simulate the production of a 20 mm deep 45° pyramidal shape by a single point incremental forming process, described in Section 6.1. The continuous bending under tension of a strip by a roll set is described in Section 6.2, where 16 cycles of forward and backward longitudinal movement of the roll set are analyzed.

6.1 Pyramidal shape

In this demonstrative application, the 45° pyramidal shape produced by SPIF is simulated. First of all, the efficient implicit method with no further enhancements is used to accelerate the standard implicit simulation. In general, a simulation is strongly influenced by its settings. Here, the influence of the increment size and the used contact model on the standard simulation and consequently on the efficient implicit method are studied. Afterwards, the achieved benefit of using the efficient implicit method is presented for two simulation settings considering the manufacturing of a 45° pyramidal shape. Finally, the advantage of combining the adaptive remeshing method and the efficient implicit method in accelerating the standard implicit simulation of the SPIF process is presented.

6.1.1 Influence of increment size

The increment size in the implicit integration procedure is limited by the accuracy requirement and the robustness of the Newton procedure (Belytschko *et al.*, 2007). In general, the size of the load increment used in the implicit time integration method is much larger than the increment size in the explicit time integration method. For an incremental forming process e.g. SPIF, modelling the sequence of small deformation increments requires thousands of numerical increments to be performed. Using too large numerical increments represents the simulation of a large number of penetrations instead of continuous incremental forming. The use of too small numerical increments within the implicit method results in tremendous computing times. Many factors influence the optimal size of the of load increment size like the FE mesh discretization and complexity of the used material model.

The size of the load increments influences the convergence of the Newton procedure and consequently the required number of iterations per increment. This has an impact on the overall CPU time required to finish a simulation. A rule of thumb can be used to define the size of the load increment in order to satisfy the requirements of SPIF process. Simply, an element has to be in contact with the forming tool for 2 or 3 load increments to model a continuous incremental forming process. This simple rule does not consider the robustness of the Newton iterative procedure that is directly influenced by the used contact description, the complexity of the material model and element type. Therefore, an automatic load increment size adjustment was developed in the used in-house FE package DiekA to adapt the increment size to the convergence behavior of the Newton procedure.

Basically, the easiness to reach convergence is used as an indicator to scale the load increment size. An easy convergence of the current load increment results in an increase of the next load increment size. If the convergence of the current load increment is not reached the current load increment is recalculated again with a smaller load increment size. To prevent a large fluctuation in the load increment size, the convergence history of a number of previous load increments is used to determine the change of the load increment size for the following increments. Also an upper limit and a lower limit is used to prevent the use of too large/small load increments.

Case study

The simulation of one loop of a SPIF process (introduced in Section 3.3.1) is used here to demonstrate the influence of the load increment size on the performance of an implicit simulation. A fixed load increment is used for the first simulation with 0.1 mm / increment in z-direction introducing a penetration of 0.5 mm and 0.25 mm / increment for the in-plane movement of the forming tool. The second simulation uses the same increment sizes with automatic adjustment of the increment size. The increment size can be enlarged 3 times and reduced 5 times.

The predicted vertical force during the simulation is shown in Figure 6.1. The fixed increment size simulation predicts large oscillation between an upper value and a lower value of the predicted force. The upper value presents the vertical loading force of the forming tool when it deforms a node to the prescribed location. Actually, the deformation is introduced by the movement of the tool over a series of nodes as shown in Figure 6.2. If the tool moves from one node to deform the next node a smooth (oscillation free) force measurement will be predicted that passes through the upper limit. This presents a continuous loading forming which is the real process. Because of the increment size, the lower limit presents the force at forming tool moving from one node to another. It shows that the numerical blank springs back because of less contact between the forming tool and numerical blank. The adjusted increment size simulation predicts almost the same upper limit of loading force as the fixed increment size simulation. It also presents a better increment size that reduces the nonphysical oscillation while the tool moves from one node to another.

The performance of both simulations is summarized in Table 6.1. The fixed increment size simulation performs 965 load increments in 5693.7 s. These increments consist of 2428 iterations and 438 line searches. For the automatic adjusted increment size simulation, the original increment size is increased by almost a factor of 3. It performs 328 load increments in 3250.4 s. The adjusted increments use 1371 iterations and 253 line searches. The CPU time is proportional to the number of the used iterations, not the used increments. Large increment sizes require large number of iterations per increment to converge, the adjusted increment simulation performs on average 4.2 iterations per increment while the fixed simulation requires 2.5 iterations per increment to converge.

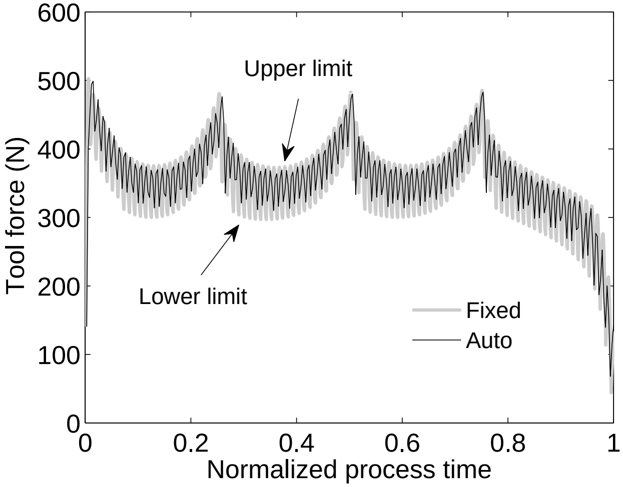

Figure 6.1: The predicted force in SPIF process by fixed increment size and automatically adjusted increment size.

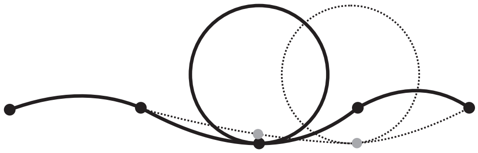

Figure 6.2: Schematic of continuous loading.

The efficient implicit method is expected to perform better with adjusted increment size simulation compared to the fixed increment size simulation because of the larger average number of iterations per increment observed in the adjusted increment simulation. The two domain efficient implicit method is used. The super elements are classified either for iterative or incremental update strategy. The 6400 shell elements are grouped in 100 super elements. A tool indicator is used with a fixed iterative ratio of 0.04. The two domain method accelerates the fixed increment size simulation by a factor of 2.26 while the adjusted increment size simulation is accelerated by 2.91 with a total CPU time of 2522.1 s and 1115.3 s, respectively. The performance of the two domain simulations is summarized in Table 6.1

Table 6.1: The simulations performance.

	Fixed increment	Adjusted increment
Nr. of increments	965	328
Nr. of Iterations	2428	1371
Iteration / increment	2.5	4.2
Nr. of line search	438	253
Newton CPU time (s)	5693.7	3250.4
Two domain CPU time (s)	2522.1	1115.3
SPEED	2.26	2.91

6.1.2 Influence of contact model

The interest in this section is to focus on the influence of contact models on the performance of SPIF implicit simulation. The penalty model and the augmented Lagrangian model are used in this thesis. Briefly, the penalty method defines the contact constraint as a multiplication of the gap function and a penalty parameter. A gap function defines the distance between possible contact nodes. The augmented Lagrangian model is a compromise between the penalty model and the Lagrangian multiplier model. In the Lagrangian multiplier model, the contact constraint is imposed in the variational equations by multiplying the gap function by a multiplier. The augmented Lagrangian model may be achieved by using an iterative update for the multiplier with a penalty-like model. The reader is referred to Zienkiewicz and Taylor (2005) for the derivations and the formulations of the models.

Defining a proper penalty parameter is a challenge. It is used in both contact models. The use of a large penalty parameter results in a stiff contact stiffness that slows the convergence of the Newton iterative procedure. A small penalty parameter results in a large penetration between the bodies in contact which reduces the quality of the achieved results. The advantage of the augmented Lagrangian model is the update of the Lagrangian multiplier similar to the linearized model in the Newton iterative procedure. This update may enhance the convergence of the contact. The update can be performed after each Newton iteration or in an added iteration loop after the convergence of the Newton iteration loop. This results in the use of more iterations.

The use of more iterations per increment in the augmented Lagrangian model may still be within acceptable limits for simulating global forming processes e.g. deepdrawing by additional hours of CPU time. In incremental forming process, for instance SPIF process, thousands of increments are performed on a relatively fine FE mesh. As discussed in previous sections, the cost of the iteration is expensive therefore performing thousands of extra iteration is going to cost additional days of CPU time.

Table 6.2: The simulations performance.

	Penalty 1 loop / 5 loops	Augmented 1 loop / 5 loops
Nr. of increments	327 / 1575	327 / 1673
Nr. of Iterations	1000 / 5028	1046 / 6965
Nr. of line search	246 / 832	238 / 2913
Newton CPU time (s)	2468.0 / 12672.6	2570.4 / 21020.4

Case study

The focus in this case study is to investigate the influence of the used contact model on the performance of SPIF implicit simulation. The Augmented Lagrangian model and the penalty model are used. The used penalty parameter is $1000\,\mathrm{N/mm}^3$. The numerical model here is similar to the used model in the previous case study except a less tight convergence criterion is used, it is 0.01. Five loops are performed, that produces a shallow 2.5 mm deep 45° pyramidal shape. The convergence criterion for the Augmented Lagrangian loop allows a penetration between the forming tool and the sheet that is less than 0.05 mm.

At the end of the first loop, it is observed that both simulations have almost the same computational performance. The augmented simulation finishes 1046 iterations in 2570.4 s while the penalty simulation performed only 1000 iterations in 2468.0 s. A significant difference is observed at the end of the five loops. The performance of the simulations is summarized in Table 6.2. The Augmented simulation CPU time is 21020.4 s for performing 6965 iterations while the penalty simulation finishes 5028 iterations in 12672.6 s. The augmented simulation requires 1973 additional iterations and 2081 additional line searches[1] to the used 832 line searches in the penalty simulation. This results in the additional overall CPU time.

The predicted tool vertical force by the contact models is shown in Figure 6.3. There are four pronounced peaks in the predicted vertical force that coincide with forming the material in the vicinity of the corner. Near the corner, the numerical blank is clamped at both edges that makes it stiffer to deform. Moving the tool from one corner to the following corner, the tool deforms material that is clamped at one side which is less stiff compared to doubly clamped sides of the blank and consequently a reduction in the predicted tool vertical force is observed. The oscillation in the predicted force is explained in Section 6.1.1 (where augmented Lagrangian model is used) by the influence of the increment size. The second reason is the used contact model hence the penalty model predicts a smoother force compared to the predicted force by the augmented Lagrangian model.

In the penalty model, an initial estimation is made that nominated a group of nodes to have

[1] The line search is a method to increase the effectiveness of the Newton method for slow convergence because of roughness of the residual force vector or deviating residual force vector from the linearized model (Belytschko *et al.*, 2007)

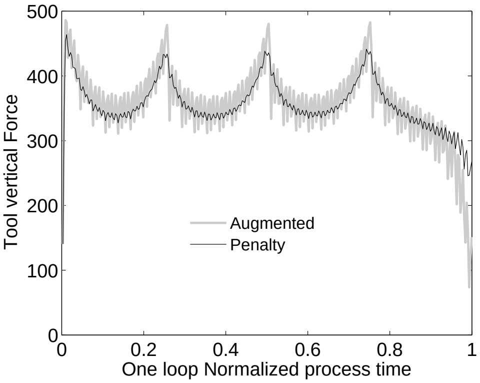

Figure 6.3: The predicted force in SPIF process by the augmented Lagrangian model and the penalty model.

contact with the tool. This initial estimation is recalculated in the augmented Lagrangian model within the iterative update of the Lagrangian multiplier and the penalty parameter. This results in finding the actual contact area that is smaller than the initial prediction. The initial prediction by the penalty model provides a large area (including more nodes). It helps in a smooth transition of the tool from a node to another (that is going to have contact in the next load increments) and consequentally in a smooth force prediction. Less nodes in contact, as predicted by the augmented Lagrangian model, results in less contact (rough transition) while the tool moves from one node to another.

At the final stage of the one loop simulation, the augmented Lagrangian model predicts a lower vertical tool force than the penalty model. The tool returns to the starting point because of the prescribed tool path of a loop, a sample of the loop tool path is shown in Figure 5.6. In the augmented simulation, the tool experiences the elastic force required to deform the material back to the initially deformed position. The tool force in the penalty simulation is higher than the one predicted by the augmented simulation because the tool has to displace the blank for a larger distance. This displacement is larger than the springback displacement and less than the sum of the springback displacement and the penetration depth in the sheet.

6.1.3 Small / intermediate numerical model

A 20 mm deep 45° pyramidal shape produced by the SPIF process is studied. The main focus is to investigate the efficient implicit method performance in accelerating the standard implicit simulation. The pyramidal shape is made of a 100 × 100 × 1.2 mm^3 initially flat blank. Two tool diameters are used: 20 mm and 10 mm. The tool diameter influences the required numerical discretization, the smaller tool requires a finer mesh. Actually, both simulations have the same ratio of element length to the tool diameter. The two simula-

Table 6.3: The simulations settings.

	First	Second
Nr. of element	6400	25600
Shell element	D. Kirchhoff T.	D. shear T.
Nr. of DOF	18729	75849
contact model	augmented / penalty	augmented
Tool diameter (mm)	20	10

tions are performed for the different tool sizes, the simulations settings are summarized in Table 6.3. 40 loops are performed using the same loop settings as in Section 5.2. A simple representative mild steel material model is used (Section 2.2.1).

The first simulation can be considered as small scale numerical model (18729 DOFs). The two domain method is used to accelerate the standard implicit simulation. The 400 super elements are classified into iterative and incremental update strategy. A simulation is performed using the tool indicator and another simulation is performed using both the tool and the plastic indicator. In general, the two domain method successfully accelerates the standard implicit simulation, the simulations performance is summarized in Table 6.4. The two domain methods achieve higher SPEED using the tool indicator only compared to the combined tool and plastic indicator. With the tool indicator only, it accelerates the standard implicit simulation by a factor of 2.73 compared to 1.80 for the combined indicators. This is a result of using a smaller iterative ratio by the tool indicator (0.141) compared to the used iterative ratio (0.358) by the combined indicator, see (4.13). The use of the augmented Lagrangian model requires more iterations to achieve the convergence compared to the penalty model, therefore a larger SPEED is achieved in the augmented based two domain simulation (2.73) compared to the penalty based two domain simulation (2.18), using the tool indicator. Almost the same number of increments, iterations and line searches are used by the two domain method compared to the standard implicit simulation.

The xz profile at $y = 0$ is used to evaluate the achieved results. The Augmented Lagrangian standard simulation has a better prediction of the xz profile compared to the penalty based standard simulation by achieving the prescribed displacement at the bottom of the pyramid that is 20 mm. The two domain method has a very good agreement in predicting the xz profile with the standard implicit simulation, this holds for both contact models. For the tool indicator based simulation, the error is limited to less than 50 μm. The use of both the tool and the plastic indicator uses a larger iterative ratio that corresponds to better prediction of the plastic zone. This results in reducing the error in the predicted xz profile to less than 15 μm

The second simulation can be classified as an intermediate scale model. The FE mesh is discretized using Discrete Shear Triangular shell elements. The FE model contains 75849 DOFs. A smaller tool diameter is used with the Augmented Lagrangian contact model. The standard implicit simulation requires almost a month (29.34 days) to finish the tool path

Table 6.4: The simulations performance of the coarse setting. The first entry belongs to augmented based simulation and the second entry is for the penalty based simulation.

	Standard	Two domain Tool	Two domain Tool + plastic
Nr. of increments	10725 / 8869	10753 / 8871	10808 / 8871
Nr. of Iterations	51322 / 31526	51142 / 31410	51457 / 31515
Nr. of line search	31521 / 2140	32308 / 2090	32634 / 2143
iterative ratio	1.00 / 1.00	0.141 / 0.141	0.358 / 0.398
CPU time (hr)	49.22 / 21.35	18.04 / 9.79	27.24 / 13.70
SPEED	1.0 / 1.0	2.73 / 2.18	1.80 / 1.55

Table 6.5: The simulations performance of the fine setting.

	Standard	Two domain
CPU time (day)	29.34	8.86
SPEED	1.0	3.31

(Table 6.5). The two domain method with the use of the tool indicator requires only 8.86 days. It accelerates the standard implicit simulation by a factor of 3.31. It has a very good agreement of the xz profile at $y = 0$ compared to the xz profile of the standard implicit simulation. The two domain method xz profile deviates by less than 60 μm, as shown in Figure 6.5

6.1.4 Two domain–adaptive refinement

The purpose of this demonstrative case study is to apply two numerical techniques simultaneously during a SPIF process simulation of a 45° pyramidal shape of 17 mm deep. These numerical techniques are the efficient implicit time integration scheme and the adaptive refinement. The adaptive refinement scheme uses the h-adaptivity method and introduces one level of refinement. The efficient implicit method splits the FE model into two domains that apply an iterative and an incremental update strategy.

The numerical blank of a $100 \times 100 \times 1.2\,\text{mm}^3$ is discretized with 3200 discrete shear triangular shell elements. It is a reference model. Each element has in total 21 integration points. A simple material model is used (Section 2.2.1), representative for a mild steel. The standard implicit time integration scheme is used to simulate the incremental forming of the reference blank into a 45° pyramidal shape by an analytical spherical tool of 10 mm radius. The simulation finishes when the tool reaches the end of loop 34, the first and the

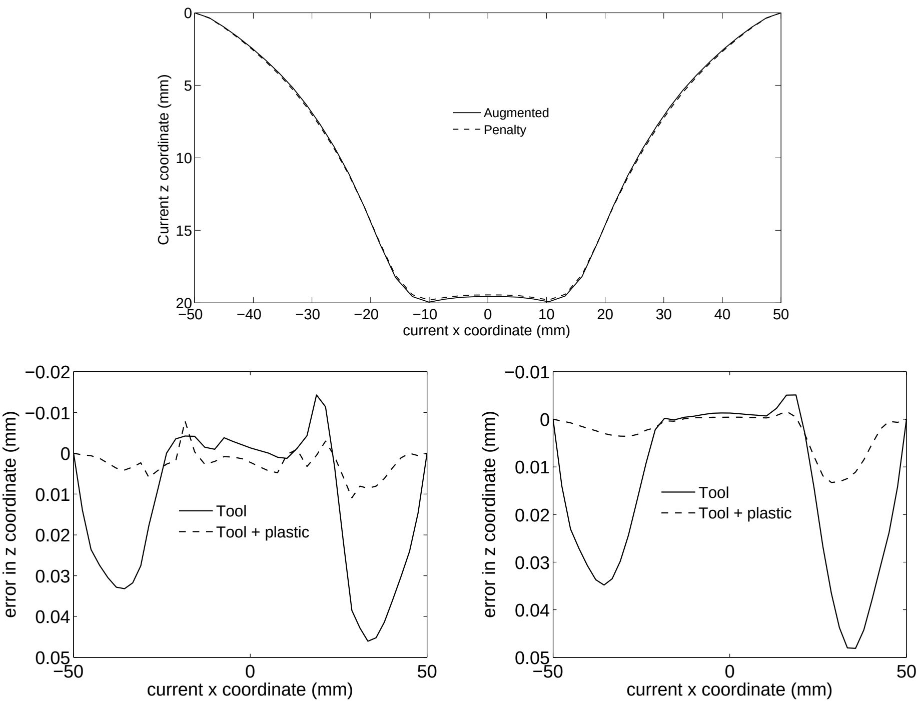

Figure 6.4: The achieved xz profile by the standard simulation using different contact models (top). The error in predicting the xz profile by the two domain method using augmented model (bottom left) and penalty model (bottom right).

final loop description is shown in Figure 5.6. Two implicit simulations are performed using an intermediate coarse initial mesh of 800 triangular shell elements. In the first simulation, the h-adaptivity introduces one level of refinement only (Refinement) while in the second simulation the adaptive refinement is combined with the two domain method (two domain–refinement). Because of adaptive refinement, the number of elements is expected to vary between 800 and 3200 within these simulations.

The overall performance of the simulations is summarized in Table 6.6. The reference simulation performs 8052 load increments in 10.54 hr. On average, the adaptive refinement method accelerates the standard implicit simulation by a factor of two, this agrees with expectation as shown in Section 5.2.1. The two domain method accelerates the adaptive refinement simulation by a factor of 1.8 resulting in an overall acceleration of the standard simulation by a factor of 3.6, performing almost the same number of increments. A high SPEED is achieved in the initial stage (the first thousand increments), the refinement simulation has a SPEED of 4 while the two domain–refinement achieves a SPEED of 8. The achieved SPEED is reduced during the simulations, the evolution of SPEED for both

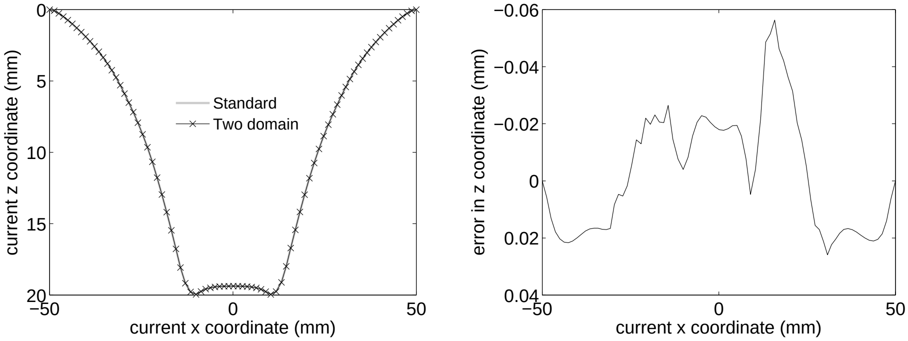

Figure 6.5: The achieved *xz* profile for the fine mesh simulation (left) and the predicted error (right).

Table 6.6: The simulations performance.

	Reference	Refinement	Two domain–refinement
Nr. of increments	8052	8145	8169
Nr. of elements	3200	800-2032	800-2048
Nr. of Nodes	1681	441-1057	441-1065
CPU time (hr)	10.54	5.27	2.92
SPEED	1.0	2.0	3.61

simulations is shown in Figure 6.6. The main cause of reducing the achieved SPEED is the increase of the number of elements. Because of the h-adaptivity method, a gradual growth of the FE model size is observed as shown in Figure 6.7. In this case study, the advantage of using the intermediate coarse mesh has a large impact on SPEED in the initial stage of the simulation. The current mesh is developed initially from an intermediate coarse mesh and its growth is limited by one level of refinement resulting in the fine mesh used in the reference simulation. The use of coarsening will not enhance the achieved SPEED significantly and it will not maintain the accuracy as discussed in Section 5.2.1.

The new elements are mainly added in the vicinity of the tool resulting in an increase of the iterative ratio as shown in Figure 6.8. The increase of the iterative ratio results in reducing the achieved SPEED of the two domain method as shown in the same figure. The refinement reduces the effectiveness of the two domain method. Here, SPEED is defined as the ratio of the required computing time by the refinement simulation to the computing time of the two domain–refinement simulation.

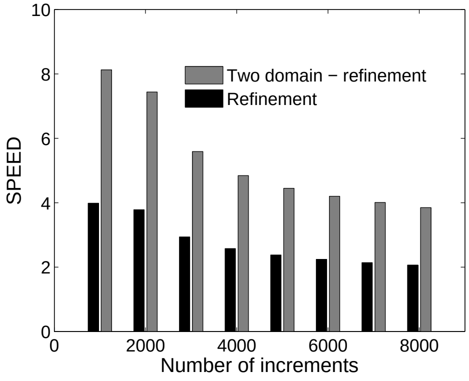

Figure 6.6: The evolution of SPEED during the adaptive refinement simulation and the two domain–adaptive refinement simulation.

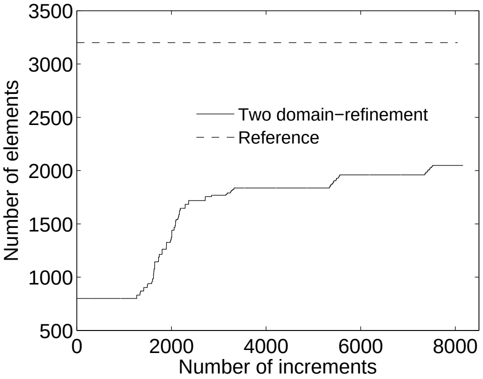

Figure 6.7: The evolution of the total number of elements in the two domain–refinement simulation, the refinement simulation has almost the same elements growth.

In general, the results achieved by the adaptive refinement approach and the two domain–refinement approach have a good agreement with the results achieved by the reference simulation. Considering the stretching strain at mid-integration point (in Figure 6.9), the maximum achieved equivalent plastic strain at the reference FE mesh is 0.487. The adaptive refinement approach predicts almost the same maximum equivalent plastic (0.496) it is overestimated by 1.8 % while it is overestimated by 1.2 % using the two domain–refinement approach which predicts a maximum plastic strain of 0.493. The observed less smooth

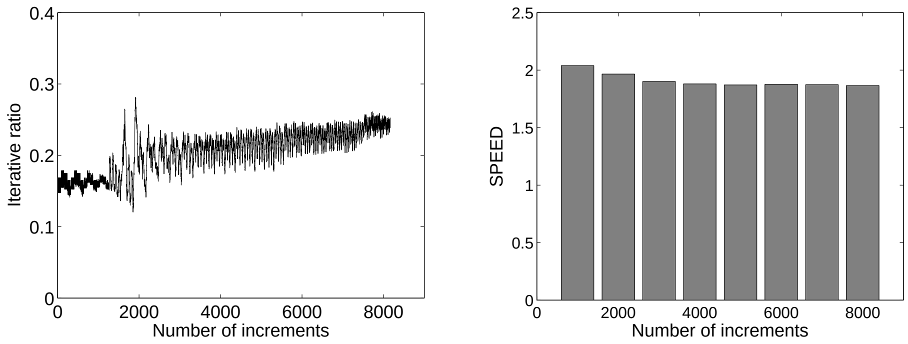

Figure 6.8: The revolution of the iterative ratio in the two domain–refinement simulation (left). The achieved SPEED of the two domain–refinement method accelerating the refinement simulation.

distribution of the result is introduced by the adaptive refinement approach. The two domain method nicely predicts even the less smooth distribution of the plastic strain.

6.2 Continuous bending under tension

The continuous bending under tension process is a multi-point incremental forming process. The first stage of the process and the setup description are explained in Section 3.3.2. The first stage introduces bending by the vertical movement of central roll near one end of the strip. Subsequently, the roll set moves to the opposite end. During this stage, both edges of the strip are fully clamped. Now, the roll set is located at the cyclic starting point position. The process proceeds by moving the far edge of the strip away to apply a tensile load. The tensile load is combined with continuous cyclic movement of the roll set. A full cycle of the roll set contains a forward movement towards the moving strip edge and backward towards the starting point as shown in Figure 6.10.

The initial bending is introduced by the vertical central roll displacement of 3.3 mm. The roll set longitudinal span movement is 100 mm traveled at 66.7 mm/s. The moving edge velocity is 2.5 mm/s. With these settings, the standard implicit simulation is performed using a penalty contact model. A two domain simulation is performed to accelerate the standard implicit simulation. The iterative and the incremental super elements are classified using a tool indicator (Section 3.3.2)

The predicted vertical force on the central roll by both simulations is shown in Figure 6.11. The process performs 16 cycles, an extensive analysis of the process is presented in the following chapter. The vertical force predicted by the two domain has a very good agreement with the force predicted by the standard simulation. The deviation is limited to a few Newtons. At the change of the roll set direction, the error is limited to less than 20 N which is observed for half of the process time. Afterwards, the deformation becomes

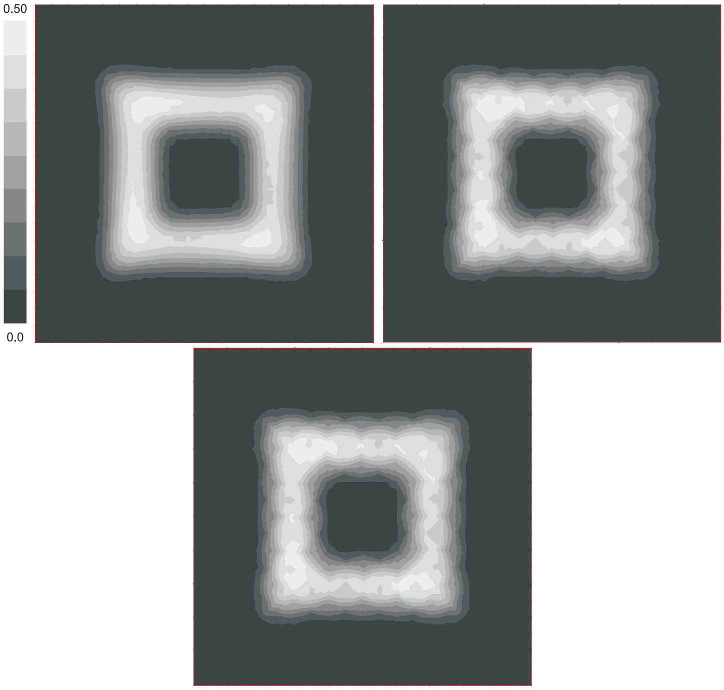

Figure 6.9: The achieved equivalent plastic strain at the mid-integration point by the reference (top left), adaptive refinement (top right) and two domain–refinement approach (bottom).

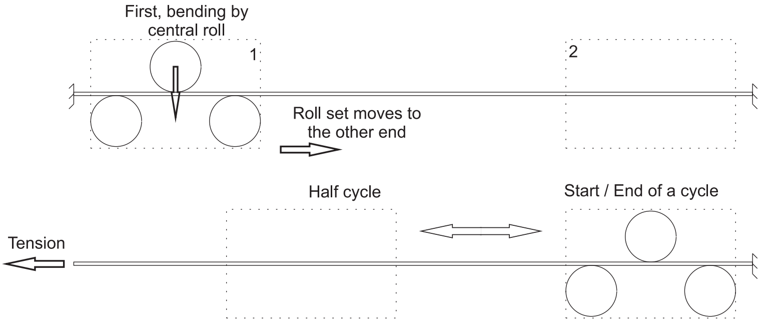
First, bending by central roll
1
2
Roll set moves to the other end
Half cycle
Start / End of a cycle
Tension

Figure 6.10: Continuous bending under tension process description.

large resulting in elongated elements (almost double the initial length) and that introduces

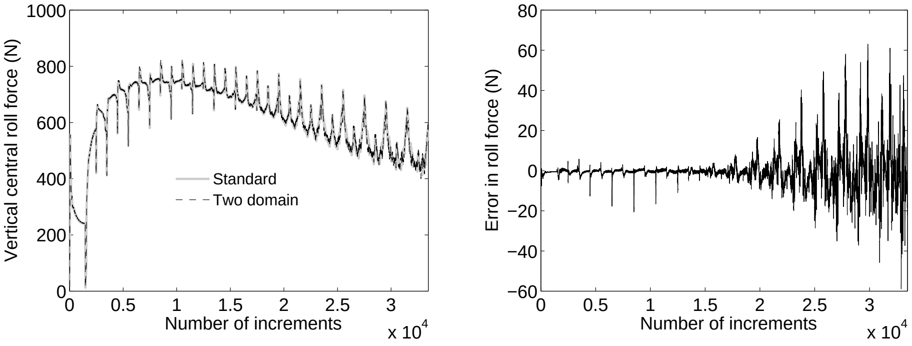

Figure 6.11: The vertical force on the central roll during CBT test (left) and the predicted error (right).

Table 6.7: The continuous bending under tension precess simulations performance.

	Standard	Two domain
CPU time (hr)	59.51	33.02
SPEED	1.0	1.8

slight oscillations in the force measurement and, of course, an increase of the error even within the cycle. The CPU time performance is summarized in Table 6.7. The standard implicit simulation finishes the 16 cycles in 59.51 hr. The two domain method accelerates the simulation by a factor of 1.8 and finishes in 33.02 hr.

6.3 Summary and conclusions

Two real-life applications of incremental forming are demonstrated. The first application is the production of a 20 mm deep 45° pyramidal shape by SPIF process, in Section 6.1. For a particular setting, the two domain method accelerates one month simulation by a factor of 3.31. It finishes the simulation in almost 9 days with an error of less than 60 μm. The achieved speeding factor for a simulation is influenced by the used contact model and the increment size because both settings influence the required number of iterations to converge. The two domain method combined with the adaptive refinement accelerates the standard implicit simulation of 3200 shell elements by a factor of 3.6. Actually, the adaptive refinement accelerates the standard implicit simulation by a factor of 2 and the two domain method accelerates the adaptive refinement simulation by a factor of 1.8.

In Section 6.2, the two domain method successfully accelerates the standard implicit simulation of the continuous bending under tension test by a factor 1.8. A very good agreement of the predicted central roll force is achieved by the two domain method simulation compared to the achieved prediction of the force by the standard simulation. A numerical investigation of the continuous bending under tension process is presented in the next chapter.

7. Continuous bending under tension

In this chapter the continuous bending under tension test is analyzed by numerical simulation. In particular the ability of achieving high strains by combined stretching and bending is considered. This deformation mode has similarities with the deformation that takes place in incremental sheet forming (ISF) and may explain the high strains that are observed there. A state-of-art for continuous bending under tension process is introduced in Section 7.1. The experimental setup and the 3D FE model are introduced in Section 7.2. Within this section, the sensitivity of the numerical model to mesh discretization is studied. Also, different material models are investigated. An isotropic hardening material model and two mixed isotropic/kinematic hardening material models are used. With satisfactory results achieved by the isotropic hardening model, a further analysis on the cyclic force–displacement curve of the CBT process is presented in Section 7.3. This analysis focuses on the pattern of the cycle that consists of two different parts and the evolution of the cycle during the process. In Section 7.4, a numerical stability analysis for inhomogeneous stress distribution is introduced. The model describes the importance of bending in stabilizing the deformation under tension. According to that criterion, a stable deformation can be achieved as long as it requires an increase of the force.

7.1 Introduction

For the last decade much research has been carried out investigating the mystery of incremental sheet forming (ISF), being that the achieved strains are often far above the forming limit curve (FLC) that is established for ordinary sheet forming. Furthermore, it has a relatively simple and cheap setup. The ISF process is limited to small volume production because of the long running time of the process. ISF is a displacement controlled process performed on a CNC machine. A clamped blank is deformed by the movement of the tool that follows a prescribed tool path (Matsubara, 1994). The process was described in a patent by Leszak (1967) without the use of a CNC machine. An extensive overview of the process is given by Jeswiet *et al.* (2005); Bambach (2008); Emmens *et al.* (2010).

Several mechanisms have been proposed in the literature to explain the increased formability that is achieved by ISF. These mechanisms are summarized and discussed in full detail in a recent review paper by Emmens and van den Boogaard (2009b). The early assumption for stabilization was based on governing through-thickness shear. This mechanism could not be confirmed by recent experiments performed using single point incremental forming (SPIF) and two point incremental forming (TPIF). Instead, stretching combined with shear in the plane perpendicular to the tool direction and shear in the plane parallel to the tool direction is proposed in the same framework (Jackson and Allwood, 2009).

Another proposed mechanism is bending under tension. It has been proposed based on 3-dimensional FE analysis (Sawada *et al.*, 2001). The deformation in ISF is subjected to bending and unbending with stretching along the meridian line and shear in the circumferential direction. Also, based on experimental works on water jet forming, a localised bending and unbending in a global stretching of the sheet is assumed to stabilize the deformation to high strain (Emmens, 2006). Very recently, a simplified continuous bending under tension (CBT) setup has been proposed to investigate the bending under tension in ISF by Emmens and van den Boogaard (2009a). In the early 70s, the basic idea of CBT was proposed to investigate the material properties at high level of straining (Benedyk *et al.*, 1971). It is shown experimentally that high levels of strain are obtained for various materials. In industry, the CBT process is implemented in the tension leveling process, which is used mainly to improve the flatness of sheets and to gain a small permanent elongation. The tension leveling process has been experimentally and numerically studied in (Mols, 1972; Yoshida and Urabe, 1999). A simple 2-dimensional FE model of the proposed CBT process has been used to study ISF by Hadoush *et al.* (2007). Indeed, the 2-dimensional FE model predicts a stable deformation up to a high level of strain.

The CBT process is, by itself, an incremental forming process. The advantages of investigating bending under tension in the CBT setup rather than on a typical ISF process are the simple stress field around the rolls and the absence of doubly curved shapes. The essentially 3-dimensional complex bending in ISF is reduced by the CBT setup to a merely 2-dimensional case. A tensile test can be carried out immediately after CBT testing. This gives the possibility to investigate the actual stress state and the achieved hardening directly, without unloading or further machining of the product.

7.2 Numerical model and process description

The experimental description of the CBT process has been explained in detail in Emmens and van den Boogaard (2009a). Within this section, some of the experimental descriptions will be mentioned for their relations to the numerical model. The CBT setup is shown in Figure 7.1. The roll set is modeled by 3 analytical, frictionless, cylinders of 15 mm diameter. In longitudinal direction, the rolls are separated from each other by 17.5 mm. The roll set can travel in the longitudinal direction only. First, the central roll is placed such as to fit the specimen in between the rolls without deforming it. Then the central roll can move in thickness direction to introduce bending. A two dimensional schematic of the FE model is shown in Figure 6.10. The bending in the specimen is introduced by the movement of the central roll downwards. The movement of the roll set in longitudinal direction introduces the bending in a cyclic manner.

The used specimen in the CBT process is schematically shown in Figure 7.2. Through the length of the specimen, the specimen has uniform thickness and piecewise uniform width. The middle of the specimen has the smallest width. The cyclic bending is performed only in the middle part of the specimen. Experimentally, it is observed that the part that experiences the combined tension and bending deforms as shown in Figure 7.3. Because of the geometry and load description, the plastic deformation of the wider parts is neglected for mild steel. The wider part of the strip can be assumed to experience rigid body motion,

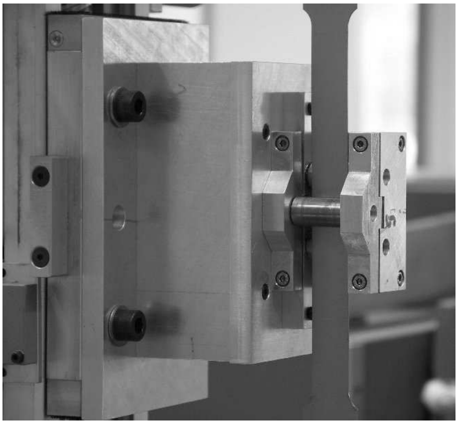

Figure 7.1: CBT setup (Emmens and van den Boogaard, 2009a).

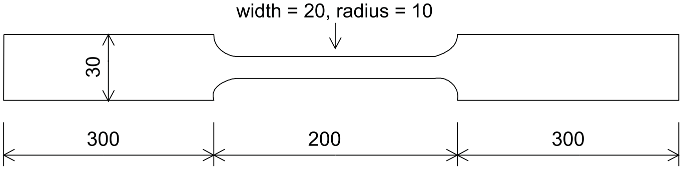

Figure 7.2: Schematic of the experiment specimen, dimension is in mm (the drawing is not to scale).

compared to the large plastic deformation in the middle part of the specimen. Only the middle part of the specimen is considered in the simulation. Because of symmetry along the longitudinal axis, half of the middle part of the specimen is modeled. The modeled part of the specimen is 200 mm in length and 10 mm in width. The thickness of the modeled part of the sheet is 1 mm.

7.2.1 Mesh dependency

Three FE meshes are used to show the influence of the FE mesh density. A regular mesh is used with 8 triangular shell elements used to discretize the 10 mm width. The triangles are large at the longitudinal symmetry line and small at the free strip edge with an element size ratio of 4 to 1. A uniform element length is used for longitudinal discretization. Three longitudinal element lengths are used and classified as: coarse (1 mm), intermediate (0.5 mm) and fine (0.25 mm). The modeled strip is imperfection-free since a nonuniform strain distribution is presented by the bending. A nonuniform strain distribution through

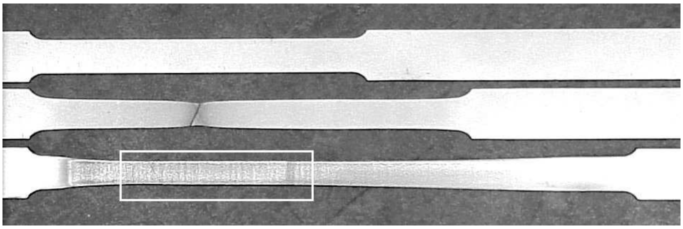

Figure 7.3: Untested and tested specimens: untested (top), tensile tested (middle) and CBT tested (bottom). A uniform deformation is observed in the white rectangle (Emmens and van den Boogaard, 2009a).

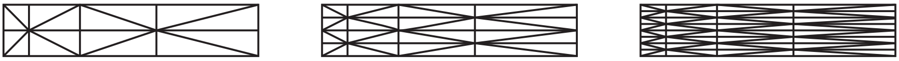

Figure 7.4: The used different mesh densities: coarse (left), intermediate (middle) and fine (right). The sample models 2 mm length and 10 mm width, with the free edge on the left.

the length is introduced by the cyclic roll set movement (Hadoush *et al.*, 2007). Samples of the different meshes are shown in Figure 7.4.

The predicted longitudinal force at the clamped edge versus the cross bar displacement is shown in Figure 7.5 for the 3 different meshes. For convenience, the absolute displacement of the cross bar will be used in this chapter keeping in mind that the cross bar travels in the negative longitudinal direction. The different meshes predict the same pattern of the force displacement curve with almost the same achieved value. The difference in the achieved predicted force value is a result of the spatial discretization. As expected, a higher level of oscillation is observed in the coarse mesh, see Section 6.1.1. The fine mesh simulation finishes 15 complete cycles and fails during cycle 16. More cycles are modeled by the coarse and the intermediate meshes. A larger element is expected to smooth the achieved strain and that results in delaying the localization of the deformation.

7.2.2 Material models

In metal forming, a yield function Φ is often used to describe the stress–strain behavior that governs the elastic–plastic mechanical behavior. The yield function Φ can be defined as

$$\Phi = \sigma_{\mathrm{eq}} - \sigma_{\mathrm{f}} \tag{7.1}$$

where σ_{eq} and σ_{f} are the equivalent stress and the flow stress, respectively. The flow stress σ_{f} defines the current yield strength, it models the size of the yield surface that may expand (hardening) or contract (softening). The equivalent stress σ_{eq} defines the shape of the yield

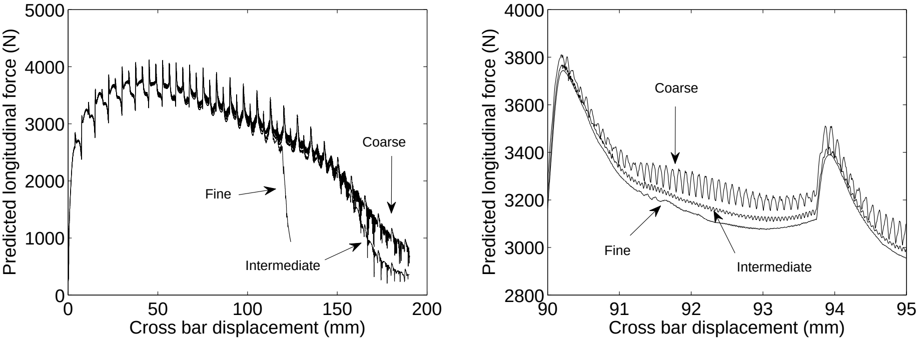

Figure 7.5: The predicted force–displacement diagrams for the entire CBT process (left) and a zoomed-in part of the process (right).

surface in stress space. For a comprehensive overview, the books by Simo and Hughes (2000); Belytschko *et al.* (2007); Hill (1950); Zienkiewicz and Taylor (2005) are useful.

In this work, the anisotropic yield function postulated by Hill (1948), also known as Hill'48 is used. In Hill'48, three orthogonal planes are presumed that lead to three principal axes of anisotropy. For sheet metal, these principal axes coincide with the rolling, transverse and thickness direction of the sheet. Hill'48 can be written as

$$\phi = F(\sigma_y - \sigma_z)^2 + G(\sigma_z - \sigma_x)^2 + H(\sigma_x - \sigma_y)^2 + 2L\sigma_{yz}^2 + 2M\sigma_{zx}^2 + 2N\sigma_{xy}^2 - x^2 \quad (7.2)$$

the value x can be scaled with the orthotropy parameters F, G, H, L, M and N. Here, the value x is equal to $\sigma_{\mathrm{f}}\sqrt{G+H}$. By this choice, σ_{f} equals σ_x in a uniaxial tensile test in the x-direction. For a plane stress condition, Hill'48 yield function is simplified by setting $\sigma_z = \sigma_{yz} = \sigma_{zx} = 0$, and becomes

$$\phi = (G+H)\sigma_x^2 - 2H\sigma_x\sigma_y + (F+H)\sigma_y^2 + 2N\sigma_{xy}^2 - 2\sigma_{\mathrm{f}}^2 \quad (7.3)$$

Isotropic hardening models a continuous expansion of the yield surface and the yield strength is equal in tension and compression. It can be modeled by the power law

$$\sigma_{\mathrm{f}} = \sigma_0 + C(\varepsilon + \varepsilon_0)^n \quad (7.4)$$

It is a simple hardening law, the four parameters can be easily fitted to a uniaxial tensile test. In cyclic loading, the isotropic hardening model provides poor results. At load reversal, the material yields earlier compared to the isotropic hardening model. This phenomenon is known as the Bauschinger effect. The kinematic hardening model of Armstrong and Frederick describes the Bauschinger effect with the use of the back stress ξ

$$\dot{\xi} = \dot{\lambda}\Big(A_{\mathrm{k}}\frac{\partial\phi}{\partial\sigma} - A_{\mathrm{l}}\xi\Big) \quad (7.5)$$

Table 7.1: DC06 material parameters.

Orthotropy parameter	F	G	H	L	M	N
Value	0.517	0.702	1.298	3	3	3.12

	isotropic	isotropic/kinematic 1	isotropic/kinematic 2
σ_0 (MPa)	0.0	0.0	95.0
C (MPa)	494	451.5	300.0
ε_0 (-)	1.0×10^{-5}	1.0×10^{-7}	1.0×10^{-9}
n (-)	0.248	0.248	0.340
A_l (-)	0.0	474	12.25
A_k (-)	0.0	9913	766.0

where $\dot{\lambda}$ is the plastic multiplier[1]. The back stress ξ stores the history of the stress path. A_k describes the hardening rate, A_l controls the contribution of the back stress in the evolution equation of the back stress. It allows a gradual increase of the flow stress after load reversal. Chaboche (1991) extended the kinematic hardening model and observed accurate results for a loading history with 10 load reversals. In (7.3), $\sigma - \xi$ is used instead of σ. The isotropic/kinematic model describes the growth of the yield surface and the Bauschinger effect for cyclic loading.

In the CBT process, a part of the strip is bent and unbent for each pass of the roll (3 times for the roll set). Three parameter sets are used in this investigation to model DC06 sheet material, the used parameter sets are listed in Table 7.1. The first set represents only isotropic hardening according to (7.4), the other two use a combination of isotropic and kinematic hardening according to (7.4) and (7.5). The latter two parameter sets are obtained by fitting with a different weighting factor for the transient zone after load reversal. Two cycles are modeled. At the end of the simulation, the element length is doubled.

The stress–strain curve for the tension–compression test is shown in Figure 7.6. A difference in the stress–strain curve is observed even for the first part of the test that introduces monotonic tensile loading. After a load reversal, the isotropic/kinematic 1 material model shows a non–sharp elastic/plastic transition compared to the sharp elastic/plastic transition that is observed in the isotropic material model. The isotropic/kinematic 2 model shows a gradual stress increase after the load reversal that provides a transient hardening effect modeling the Bauschinger effect.

An experiment is performed with a roll speed of 66.7 mm/s and cross bar velocity of 2.5 mm/s. Bending is introduced by shifting the central roll 3.3 mm in thickness direction and it is held at this level for the entire process. The predicted force at the clamped edge for each material model is plotted versus the cross bar displacement in Figure 7.7. In general, the force–displacement curves predicted by the three different material models are

[1] Drucker's postulate requires that the plastic strain is perpendicular to the yield surface: $\dot{\varepsilon}^\mathrm{p} = \dot{\lambda}\frac{\partial \sigma_\mathrm{eq}}{\partial \sigma}$

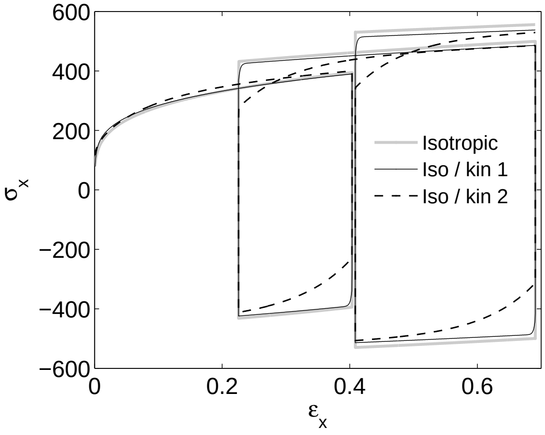

Figure 7.6: Stress–strain curves for the components in x-direction.

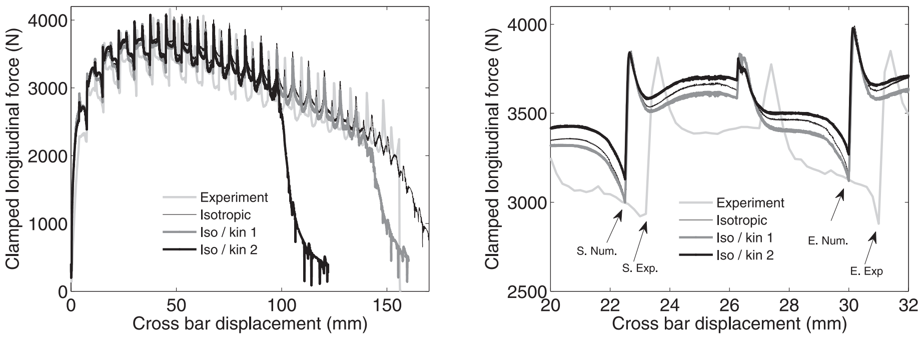

Figure 7.7: Force–displacement curves for different material models: the entire process (left) and one cycle (right).

very similar to the experimental force–displacement curve. The different material models predict successfully the sudden increase of the force at the start and at the middle of the cycle. At the second half of the cycle, the material models predict the gradual decrease of the force at the final stage of the cycle. Astonishingly, the Bauschinger effect that is included in the isotropic/kinematic material models shows no significant difference in the pattern of the predicted force compared to the predicted force by the isotropic material model. However, the isotropic/kinematic material models show earlier localization than the isotropic material

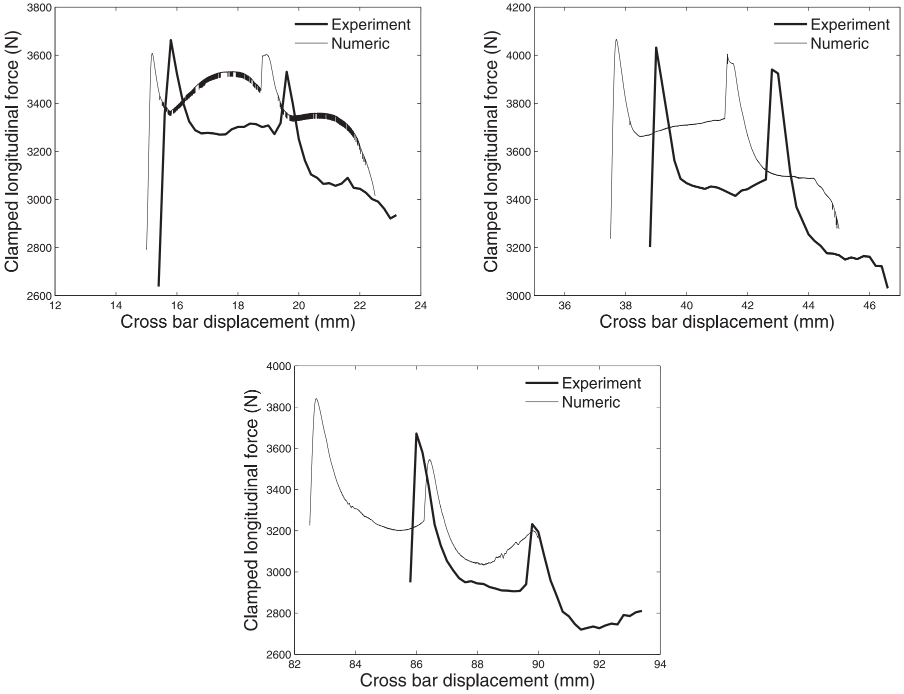

Figure 7.8: Evolution of the cyclic force–displacement curve during the CBT process: the third cycle (top left), the sixth cycle (top right) and the twelfth cycle (bottom).

model. This contradicts with the expectation that kinematic hardening models can stabilize the process because of the higher slope of the stress–strain curve compared to the slope in the isotropic material model as can be seen in Figure 7.6.

7.3 Force displacement curve

The purpose of this section is to analyze the force–displacement curve. In recent work of Emmens and van den Boogaard (2009a), the force–displacement curve is presented for the CBT process, three representative cycles of the force–displacement curve are plotted in Figure 7.8. They explained some aspects of the force measurement on the CBT process. The peaks (the sudden increase of force) were explained by the deceleration of the roll set to change the movement direction resulting in an increased tensile force. The additional positive/negative contribution of the down/up movement of the roll set was used to explain the change in average of the force between the first and the second half of the cycle.

However, the numerical model does not include the acceleration/deceleration of the

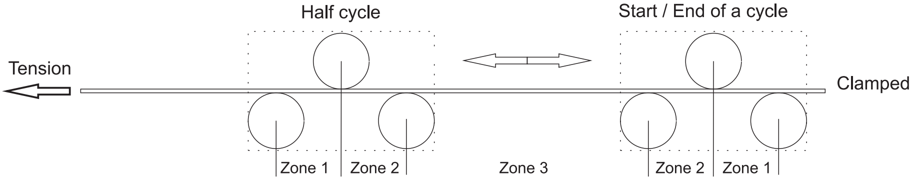

Figure 7.9: Different deformation history within the cyclic roll set movement. The number of the zones defines the number of bending/unbending operations.

roll set, instead it changes the direction of the roll set immediately and the peaks are still observed. Ignoring the acceleration/deceleration of the roll set results in predicting a smaller span length of cycle in the FE model compared to the experimental span length of the cycle. This explains the lag of the measured force compared to the predicted force. It is true that the roll set has positive/negative contributions to the clamped longitudinal force but it does not explain the evolution of the shape of the cyclic force–displacement curve during the process.

Within the continuous bending under tension process, the bent material portions deform plastically. Because of the bending contribution they require less tension force to deform compared to the force required to deform the same cross section under tension only (Marciniak and Duncan, 1992). Therefore, the rest of the strip material is loaded elastically. Due to the geometry of the roll set (three identical rolls in a row) the strip is bent and unbent at different levels. Three zones are defined based on the frequency of bending/unbending as shown in Figure 7.9. In zone 1, the material is bent and unbent once for each pass of the roll set. Similarly, the material experiences 2 sets of bending/unbending and 3 sets of bending/unbending in zone 2 and zone 3, respectively. For simplicity, the zones are defined from one roll center to another roll center. Zone 1 and zone 2 have span length of 17.5 mm which equals the longitudinal distance from one roll to another. Zone 3 has the largest length, it is equal to the span length of the cycle (here it is 100 mm) minus the distance between the lower rolls (35 mm).

The strip is under tension during the CBT test because of the cross bar displacement. This results in dragging the material to the left as shown in Figure 7.9. Consequently, the material will migrate from one zone to another that has a different frequency of bending/unbending. Eventually, part of the material that experienced bending leaves the bending/unbending zones to be under tension force only as the rest of the strip. The different zones of bending/unbending frequency and material migration through these zones are process characteristics of the CBT test. They are the keys to explain the evolution of the shape of the cyclic force–displacement curve.

7.3.1 Cycle

The cycle consists mainly of two peaks and two almost steady parts. The peaks are observed at the interval when the roll set changes its travelling direction. The steady parts present the major part of the cycle. The first half of the cycle models the roll set travelling towards the moving cross bar and away from the clamped edge (where the force is measured). Therefore, the measured force on average is higher in the first half of the cycle compared to the measured force in the second half of the cycle. With the use of an isotropic material model only as shown in Figure 7.8, the numerical model predicts the cyclic force–displacement curves in very good agreement with the experimentally achieved cyclic force–displacement curves.

Steady state

The goal of this discussion is to explain the evolution of the steady state parts of the cycle. The longitudinal force–displacement cyclic pattern depends on two process characteristics: the different bending/unbending zones and the migration of the material toward the moving cross bar. These two factors result in different levels of hardening along the strip and consequently thickness distribution. The hardening and the thickness distribution along the strip implicitly influence the evolution of the cyclic force–displacement curve. To simplify the discussion, the force–displacement curve is first explained based on the different bending/unbending zones, then the additional influences of the other factors are introduced.

Within the travelling distance of the roll set and because of the roll set geometry, three different bending /unbending zones are distinguished. The cyclic forward / backward movement of the roll set shifts the active location of the plastic deformation through these zones assuming that only the bent material deforms plastically. An initial assumption of the cyclic force–displacement curve is plotted in Figure 7.10. This assumed curve considers the use of an isotropic hardening material model and the positive/negative contribution of the longitudinal force of the roll set. Also, it assumed a uniform deformation within each zone. At the beginning of the cycle, the roll set is located in zone 1 and zone 2, the zone locations are shown in Figure 7.9. Zone 1 and zone 2 are less hardened than zone 3 because they are less deformed. The roll set moves toward the cross bar and gradually it deforms more material located in zone 3, all material in zone 2 and less material in zone 1. This results in an increase of the force until the complete shift of the roll set to zone 3. This requires the roll set to travel 35 mm which is the length of zone 1 and zone 2. Then, a constant force is achieved because a uniform deformation is assumed within the zone and the roll set is completely in zone 3. The span of the cycle is 100 mm, limiting the length of the steady force measurement to 30 mm. Then, a decrease of the force is observed as the roll set deforms material in zone 2 and subsequent material in zone 1. At the end of the first half of the cycle, the roll set is located in zone 1 and zone 2 near the cross bar. In the second half of the cycle, the roll set moves backward to the clamped edge of the strip. The second half of the cycle is assumed to be a mirror of the first half except a lower level of the constant force part is assumed considering the negative contribution of the roll set. The following cycle will be performed at a higher level of force compared to the force level of the current cycle because of hardening.

Up to this point, each half of the cycle consists of an increase of the force then a constant level followed by a decrease of the force. The numerically achieved results of the longitudinal force for the different cyclic patterns are plotted versus the absolute displacement of roll set in Figure 7.11. The observed pattern of the third cycle and the second half of the sixth cycle have a very good agreement with the assumed cyclic pattern based on different

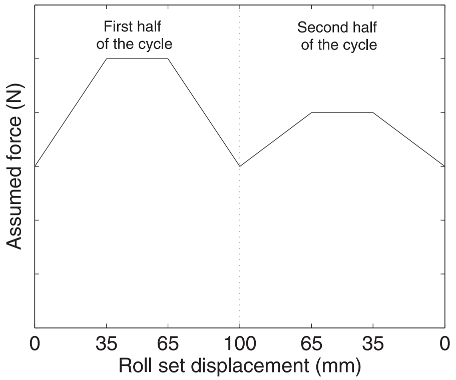

Figure 7.10: Assumed force–displacement curve for isotropic hardening material. For convenience, the absolute roll set displacement is used, the roll set travels 100 mm toward the moving cross bar (negative direction) in the first half of the cycle and 100 mm backward to the clamped edge in the second half of the cycle (positive direction).

bending/unbending zones. Still, some aspects of cyclic evolution are not explained, specially the pattern of the first half of the sixth cycle dominated by a slight positive slope, the concave-like pattern observed in the twelfth cycle and the peaks that appear after a travelling direction change of the roll set.

Because of the moving cross bar, the material is dragged toward the cross bar. The material migrates from one zone to another in one direction toward the cross bar. To emphasize the influence of material migration, the history of several nodes through the length of the strip is shown in Figure 7.12. Clearly, node 1 hardly moves (less than 1 mm) for 12 cycles. This means that material in zone 1 near the clamped edge has the lowest cyclic history and consequently is less hardened. Significantly, material migrates from zone 3 and gradually fills zone 2 and subsequently zone 1 near the cross bar. For the first half of the sixth cycle, zone 2 and zone 1 are completely filled with a material originally deformed in zone 3. This explains the almost constant level of force (a slight positive slope of the force is observed). Of course, the sixth cycle finishes with a decreasing force because the roll set goes back to zone 2 and zone 1 near the clamped edge.

Performing more cycles, material keeps migrating. Eventually, some of the material that is originally deformed in zone 3 leaves the bending/unbending zones. The material (between node 3 and node 5) has the highest level of hardening and consequently has the most significant reduction of thickness along the strip. This material, because of the significant reduction of the cross section, requires relatively less force compared to the force required to deform materials in zone 2 and zone 1 resulting in the concave-like pattern of the twelfth cycle. The hardening is a material property. Using another material with different work hardening rate influences the development rate (fast/slow) of the cyclic pattern from one pattern to another, but it will never change the process characteristics.

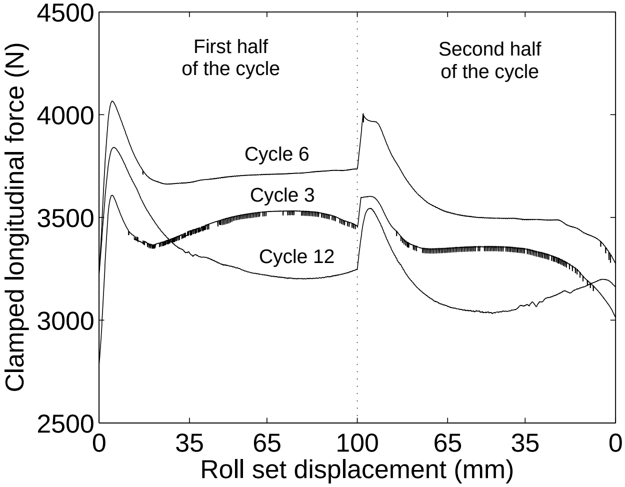

Figure 7.11: Cyclic force evolution versus the absolute displacement of the roll set.

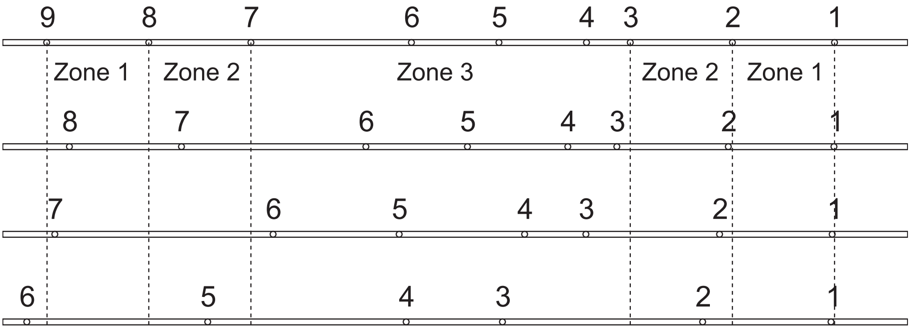

Figure 7.12: Migration of material from one zone to another toward the moving edge: initial location (top) then at the beginning of the third cycle, the sixth cycle and the twelfth cycle(bottom).

Peak

As mentioned earlier, the peak in the force during the CBT process is a sudden increase of the longitudinal force. It is observered when the roll set changes its travelling direction at the beginning of the first half and the second half of the cycle. It has been suggested

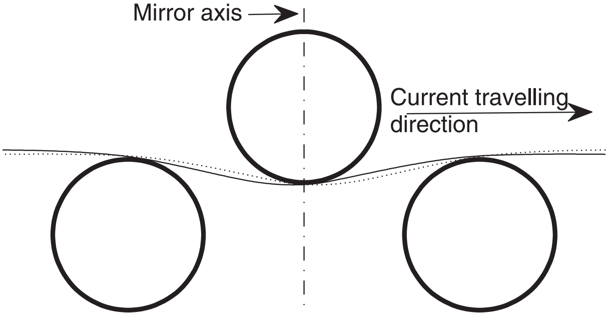

Figure 7.13: The local geometry of the strip in the FEM model (solid line) just before reversing the travelling direction of the roll set. The dotted line is a mirror image of the current geometry of the strip.

previously (Emmens and van den Boogaard, 2009a) that the peak is a result of the acceleration/deceleration of the roll set in the experiment. But in the FE model, the roll set changes its travelling direction instantaneously at the same speed. Even then, the peak is observed, it lasts for a significant interval of the roll set travelling distance and the response is not instantaneous. Up to now, the different zones of bending/unbending frequency and material migration give no direct explanation of the peak. In this section, further analysis is carried out to explain the peak.

During the steady state part of the cycle, the active plastic zone is shifted with the travelling roll set. Actually the travelling roll set shifts the local curvature and as a result the active plastic zone is shifted. During the peak, an increase of the longitudinal force (far from the assumed force expectation) is observed. This raises the following question: does reversing the travelling direction of the roll set result in a sudden change in the local curvature? The local geometry of the strip in the FEM model just before reversing the roll set travelling direction is plotted in Figure 7.13. This is the current geometry at the end of the fourth cycle, it is used to investigate the peak at the beginning of the fifth cycle. The current travelling direction of the roll set is toward the clamped edge (from left to right) and it is going to reverse. A mirror image of the current geometry clarifies that the roll set can travel relatively easier to the left than to the right. Furthermore, reversing the travelling direction of the roll set (moving to the left) will not result in a sudden and significant change in the geometry of the strip. This geometry will last for an interval of time.

With the movement of the cross bar, the strip must be elongated, but a balance has been reached recently between the tensile force and the bending moment (the current curvature) and the roll set needs time to change the curvature of the strip after the reversal. Therefore, the force is increased to elongate the strip compensating the cross bar incremental displacement. As a result of the increased tension force the strip plastically deforms in the region of the roll set under bending and tension and partially under tension away from the roll set. In Figure 7.14, the plastic strain per increment is plotted for two positions of the roll set, one

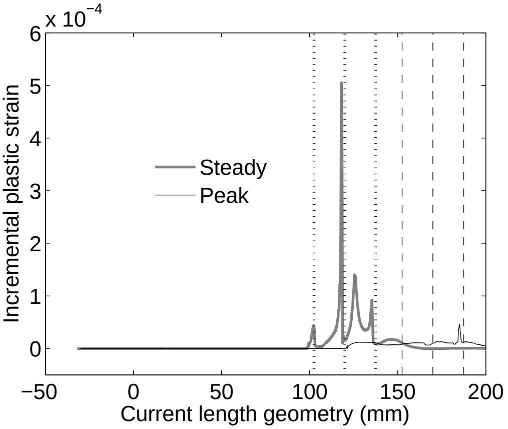

Figure 7.14: A comparison of the plastic increment for two positions of the roll set, one during the steady part (dotted lines) and one just after the reversal of the travelling direction (dashed line).

during the steady state (dotted lines) and one just after the travelling direction reversal of the roll set (dashed line). Because of the deformation, the current length of the strip at the beginning of the fifth cycle is 230 mm. It can be clearly seen that during the steady state high plastic strain increments appear, but only near the position of the rolls. Just after the reversal of the travelling direction, the peak plastic strain increments are much lower but they extend over a larger region.

The increase of the longitudinal force when the roll set reverses its travelling direction is a result of a continuous movement of the cross bar. So, what will happen if the cross bar does not move at this stage? To understand the influence of the continuous movement of the cross bar during the reversal of the roll set direction, a study on the force measurement is carried out on the first peak of the fifth cycle. The study compares the longitudinal force during the reversal of the roll set direction with and without the movement of the cross bar. The comparison of the forces is shown in Figure 7.15. Without the movement of the cross bar, the roll set travels for 5 mm at almost constant force (a slight drop of the force is observed at the very beginning) with no significant change in geometry. Then, the roll set further movement corrects the geometry and the bending/unbending starts producing significant plastic deformation (the strip is elongated) which reduces the tension in the strip. This supports the claim that the peak results from the deformation of the strip by increasing the tension force (caused by the continuous movement of the cross bar) with no significant change in strip curvature.

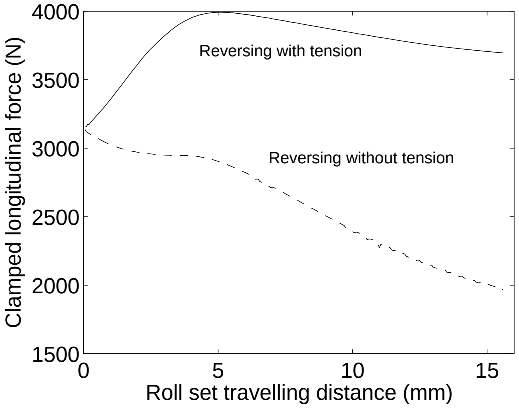

Figure 7.15: Longitudinal force evolution for the first part of the fifth cycle with and without cross bar movement.

7.4 Stability analysis

In general for bending under tension, the longitudinal stress σ_x is inhomogeneous through thickness because of bending. This is the crucial difference with the standard stability analysis of an ordinary tensile test. Depending on the tensile load level, the inhomogeneous stress distribution σ_x has a positive part and it can have a negative part. Through the width, the strip deforms approximately in a plane strain condition at the symmetry line along the length and in uniaxial condition at the free edge. Because of inhomogeneity, the instability criterion for bending under tension cannot be derived analytically. Therefore, it is derived numerically for the CBT process.

In the steady state part of the cycle of the CBT process, the material portions that experience combined bending and tension show local plastic deformation, it is located in the vicinity of the rolls. A cross section, related to such material portions, is deformed from its former geometry to the current geometry under simultaneous bending and tension. In this work, a study is carried out investigating whether additional tension (perturbation) to the current equilibrium between bending and tension for a cross section will result in a stable or instable deformation. A stable increase of the local length Γ requires an increase of the longitudinal force. This means that the rate change of the longitudinal force with respect to the local length[2] has to be positive

$$\frac{\mathrm{d}F_x}{\mathrm{d}\Gamma} > 0 \tag{7.6}$$

The tensile force is obtained by integrating the longitudinal stress σ_x over a cross section

$$F_x = \int_A \sigma_x \mathrm{d}A = \overline{\sigma}_x A \tag{7.7}$$

[2] In this section, x represents a coordinate along the curved strip in longitudinal direction

where $\overline{\sigma}_x$ and A are the average stress through a cross section and the cross sectional area, respectively. The average stress is used for convenience in the following derivation. Substituting (7.7) into (7.6) results in

$$A\frac{\mathrm{d}\overline{\sigma}_x}{\mathrm{d}\Gamma} + \overline{\sigma}_x\frac{\mathrm{d}A}{\mathrm{d}\Gamma} > 0 \tag{7.8}$$

Neglecting the elastic deformation and considering constant volume during deformation, the initial geometry deforms to

$$V = A_0\Gamma_0 = A\Gamma \tag{7.9}$$

the infinitesimal length strain increment of the mid-plane can be written as

$$\mathrm{d}\varepsilon_x = \mathrm{d}\Gamma/\Gamma_0 = -\mathrm{d}A/A \tag{7.10}$$

with the use of (7.10) and rearranging terms in (7.8), the stability condition (7.6) can be written as

$$\frac{\mathrm{d}\overline{\sigma}_x}{\mathrm{d}\varepsilon_x} > \overline{\sigma}_x \tag{7.11}$$

where $\mathrm{d}\overline{\sigma}_x/\mathrm{d}\varepsilon_x$ is the average tangent stiffness.

A stable increase of the local length Γ requires that the average tangent stiffness $\mathrm{d}\overline{\sigma}_x/\mathrm{d}\varepsilon_x$ must be larger than the average stress $\overline{\sigma}_x$. The perturbation (increase of force) shifts some of the fibers from compression to tension resulting in increasing the average stress $\overline{\sigma}_x$. The most important fact is that the average tangent stiffness $\mathrm{d}\overline{\sigma}_x/\mathrm{d}\varepsilon_x$ is much higher than the average stress $\overline{\sigma}_x$ as long as there are still elastic fibers. For a strain hardening material, the average tangent stiffness can be integrated over the cross section that has a thickness t per unit width as

$$t\frac{\mathrm{d}\overline{\sigma}_x}{\mathrm{d}\varepsilon_x} = \int_{e1}^{t/2}\frac{\mathrm{d}\sigma_x}{\mathrm{d}\varepsilon_x}\mathrm{dz} + \int_{e2}^{e1}\frac{\mathrm{d}\sigma_x}{\mathrm{d}\varepsilon_x}\mathrm{dz} + \int_{-t/2}^{e2}\frac{\mathrm{d}\sigma_x}{\mathrm{d}\varepsilon_x}\mathrm{dz} \tag{7.12}$$

where $e1$ and $e2$ define the boundaries of the elastic zone through the thickness. The first and the third term in (7.12) present the contribution of the plastically loaded fibers in tension and compression in the average tangent stiffness, respectively, while the second term presents the elastically loaded contribution in the average tangent stiffness. In the limit of a rigid perfect plastic material model, the average tangent stiffness $\mathrm{d}\overline{\sigma}_x/\mathrm{d}\varepsilon_x$ still has a high value because the second term in (7.12) is integrated over zero thickness with infinite stiffness. In the initial work of Hadoush *et al.* (2007), the compressive stress has been proposed to stabilize the deformation in the CBT test but it was not quantified while in Emmens and van den Boogaard (2009a) it has been quantified using a rigid perfect plastic material model ignoring the thickness change, resulting in a zero right-hand side in (7.11). Equation (7.11) includes the thickness change and presents a general criterion for stability in the CBT test and equally well in the tensile test.

To validate the stability hypothesis in the FEM analysis of the CBT process, the history of the failed element is tracked. For a big picture, the clamped longitudinal force for the last cycles is plotted in Figure 7.16, the stresses are plotted in Figure 7.17 and an elastic indicator is plotted in Figure 7.18. The clamped longitudinal force clearly shows that unstable deformation governs cycle 17 and it starts at the end of cycle 16 when the force drops instead of increasing like the end of cycle 15. For the same interval, σ_x for the lower

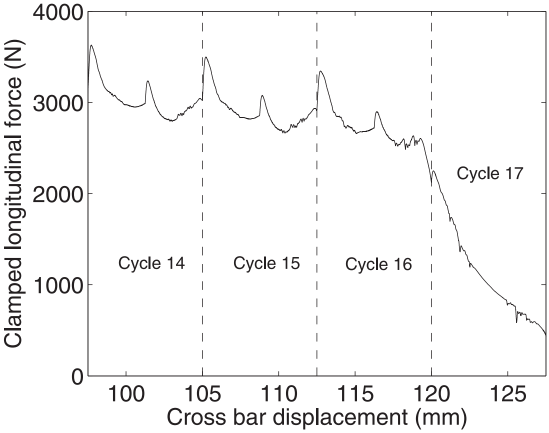

Figure 7.16: Force–displacement curve of the last four cycles for CBT process using the fine mesh.

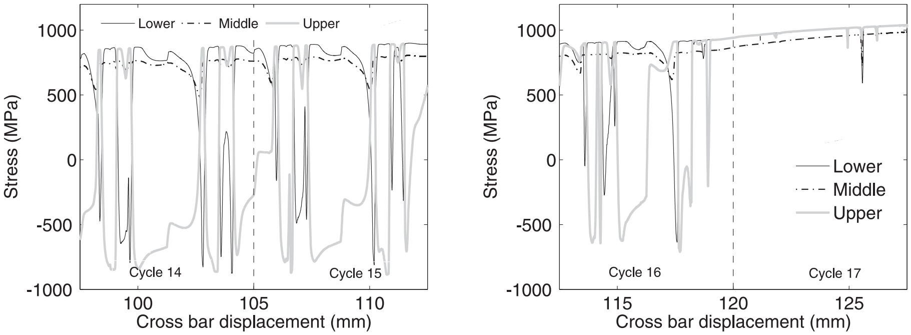

Figure 7.17: The evolution of σ_x through thickness for the failed element at the symmetry line.

integration point (first), middle (fourth) and the upper (seventh) are all in tension. σ_x varies for the rest of the integration points between the upper stress and the lower stress, they are not plotted to keep the figure clear. The middle point stress is always in tension and the upper stress and the lower stress vary between compression and tension during the pass of the roll set. When the roll set is away, the upper point stress σ_x is in compression and σ_x at the lower point is in tension.

At the beginning and at the end of cycle 15, σ_x for the lower and the upper integration points are in tension. Now, an elastic indicator is used to check whether the average tangent

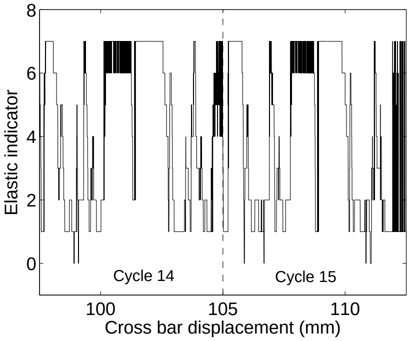

Figure 7.18: The evolution of the elastic indicator for the failed element at the symmetry line.

stiffness has a high or a low value. Each integration point is weighted by one if it is elastically deforming and by zero if it is plastically deforming. The elastic indicator for one column of integration points in thickness is the sum of the weights. Using seven integration points through the thickness, a column of integration points is fully elastic if the elastic indicator equals 7 and fully plastic if the elastic indicator equals 0. If the elastic indicator varies between 1 and 6, it deforms in a mixed combination of elastic/plastic deformation. The elastic indicator shows that for a small interval in both cycles 14 and 15 a full plastic deformation through the thickness is observed. With the help of the stress history, it becomes clear that within these intervals σ_x for all integration points is in tension. Cycle 13 is the last stable cycle based on the elastic indicator and the compressive stress, the evolution of the stresses and the elastic indicator for this cycle are plotted in Figure 7.19. Now, the question is why the cross bar incremental displacement did not localize in the failed element during these intervals or in other words why the force–displacement curve does look stable for cycle 14 and 15?

The presented results for compressive stress and elastic indicator concern only one column of integration points related to one integration point in-plane and as it was shown it violates the stability hypothesis. The structural response for a cross section is the sum of the response of many integration points in the considered cross section. The structural response for a cross section is unstable when the entire cross section is plastically deforming under tension. The presented results are for the first integration point in-plane that violates the hypothesis and it informs us that the failure of the cross section will be in the following cycles. For defect-free final products, this is important information that recommends stopping the process at the end of cycle 13. Unfortunately, the predicted stability is mesh dependent, for instance further stable cycles are performed using coarser mesh. Using the intermediate mesh discretization (shown in Figure 7.4), cycle 15 is performed without violation of the stability hypothesis for the weakest integration point. The numerical analysis is used here to investigate the underlying mechanisms of stability in the CBT process, not to predict the actual maximum achievable strain. Both mesh descriptions show that the criterion is

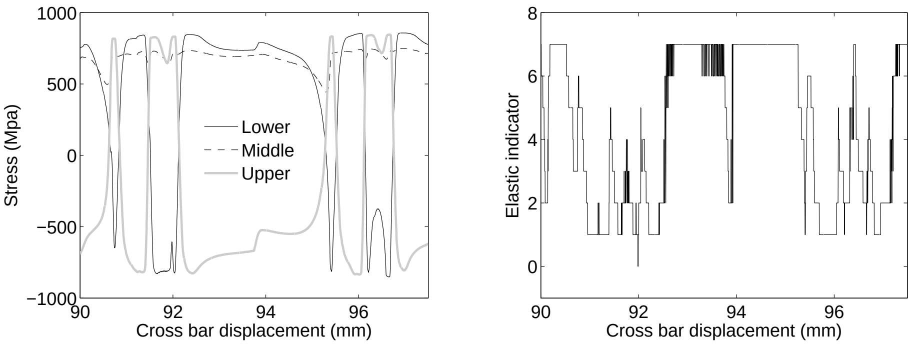

Figure 7.19: The evolution of the σ_x through thickness (left) and elastic indicator (right) for the failed element at the symmetry line during cycle 13.

violated at the first peak of the cycle, an increase of tensile force is observed during the peak. This is validated by the experimental observation that the strip mostly fails at the beginning of the cycle.

7.5 Summary and conclusions

The continuous bending under tension (CBT) process is an incremental forming process. The advantages of investigating bending under tension in the CBT setup rather than on a typical ISF process are the simple stress field around the rolls and the absence of doubly curved shapes. The essentially 3-dimensional complex bending in ISF is reduced by the CBT setup to a merely 2-dimensional case. Using the CBT process, the ability of achieving high strain by combined stretching and bending is investigated.

A 3-dimensional FE model is created for the central zone in the specimen that is plastically deforming. The model is discretized by triangular shell elements based on discrete Kirchhoff theory. Three different meshes are investigated: coarse, intermediate and fine. One dimensional refinement is considered for the element size in the longitudinal direction. The used different meshes show no significant difference in predicting the force–displacement curve of the process during the stable deformation. A higher level of numerical noise is observed in the predicted force displacement curve using the coarse mesh compared to the predicted force–displacement curve achieved by the fine mesh. Early prediction of specimen failure is observed using the fine mesh discretization as presented in Section 7.2.1.

In Section 7.2.2, three different material models are used to model the CBT process. These models are one isotropic hardening model and two isotropic/kinematic models. Despite the significant difference between these models in predicting the stress–strain curve after load reversal, only a slight difference between the models is observed in predicting the force–displacement curve. The force–displacement curve achieved by a simple isotropic hardening model has a very good agreement with the experimentally measured force–displacement curve. A major difference between the material models is observed in predicting the failure of the specimen.

A further analysis of the force–displacement curve for the CBT process is described in Section 7.3 based on an isotropic material model. Focusing on the process description, the cyclic force–displacement curve consists of two parts: a steady part and a transient part (peak). The steady part of the cycle models the deformation of the strip governed by significant curvature change of the strip because of bending. The peak results from the deformation of the strip by increasing the tension force with no significant change in strip curvature. The peak is observed twice during the cycle after reversing the traveling direction of the roll set. During the process, the cycle pattern develops. This development is a result of several factors. The main factors are: different zone of bending/unbending frequency and material migration through these zones. These factors produce different levels of hardening through the length of the strip and consequently a thickness distribution. The hardening and thickness distributions implicitly influence the development of the cycle pattern.

A numerical stability criterion is derived for inhomogeneous stress distributions through the cross sectional area in Section 7.4. The model describes the importance of bending in stabilizing the deformation under tension. A stable deformation can be achieved as long as it requires an increase of the force. Irrespective of plastic hardening, a relatively large change in the force occurs if a part of the cross section is still elastic, or for a rigid plastic model, if a part is still in compression.

8. Conclusions and Recommendations

In this thesis, an efficient implicit method is introduced to accelerate the standard implicit FEM simulation of single point incremental forming (SPIF). The method is explained in several chapters focusing on the basic concept of the method, implementation, performance and its extendability. Additionally, a fundamental study on the mechanics of a bending dominated incremental sheet forming process is presented.

Basic concept

A study on the evolution of nonlinearity during the calculation of SPIF by an implicit time integration scheme concludes that the standard use of the scheme is inefficient. A small part of the system of equations experiences a strong nonlinearity (a combination of geometrical and material nonlinearity) and requires a fully nonlinear update procedure. The rest of the equations, a large part, experiences only weak geometrical nonlinearity which does not require the expensive nonlinear iterative procedure.

An efficient implicit time integration scheme is introduced based on a mixed update procedure. Within an increment, the strong nonlinearity has a full nonlinear update treatment. The weak nonlinearity has a pseudo-linear update treatment and a nonlinear update is applied only at the beginning of the increment to include the previous nonlinear history. After that, the predictor is reused and the corrector is linearly updated. It is demonstrated by several case studies (with an acceptable margin of errors) that the efficient implicit method is as accurate as the standard implicit method and accelerates the standard implicit method by a factor of $2-3$. The speeding up is a result of the relative cheapness of the pseudo-linear treatment compared to a full nonlinear treatment of a large part of the model.

Optionally, the pseudo-linear treatment can be applied for a group of increments, which increase the achieved speeding factor. This gives the possibility to divide the FE model into three domains. The first domain has an iterative update treatment and the pseudo-linear domains that use the incrementally and the multi-incrementally update treatment. In the penetration test, it is observed that the achieved speeding factor of the three domain method is 20% higher than the achieved speeding factor of the two domain method. A better performance of the three domain method is expected for a larger blank.

Implementation

The standard implicit scheme can be easily adapted to include the efficient implicit method. The use of super elements facilitates the partitioning of a FE model into different domains with adapted update frequencies. Because of the localized plastic deformation in SPIF, a strong nonlinearity is observed in the vicinity of the tool that travels following a tool path, it is a small traveling plastic zone within a large elastically deforming environment. In order to define a proper distribution of the domains, several indicators are developed to generically classify the super elements. These indicators are the current tool location, plastic deformation in the previous load increment and the shape change in the previous load increment. The tool indicator uses a search radius to classify the domains, it requires experience to use (rule of thumb). The plastic indicator is used to shift incrementally updated super elements into the iterative update treatment. It enhances the prediction of the assumed plastic zone. The geometrical indicator is dedicated to follow the shape change of the multi-incremental super element, if it is too large, the super element is re-classified to incremental update treatment.

Performance

The speed factor (SPEED) measures the efficiency of the super element based efficient implicit approach in accelerating the standard implicit simulation of localised deformation processes. SPEED is defined as the CPU time cost of one calculation increment of a standard algorithm compared to the cost of one increment of the efficient implicit time integration procedure. SPEED is influenced by several factors. These factors are the number of performed iterations, the combination of the different update strategy ratios, the used update strategies and the cost of the major parts of the Newton iteration (BUILD, UPDATE and SOLVE). The major parts ratio of a Newton iteration depends on the material model and the element type.

An analytical formula for the two domain method made of iteratively and incrementally updated super elements is developed. It shows that SPEED is enhanced by reducing either the SOLVE ratio or the iterative ratio (the ratio of iteratively updated super elements to the total number of super elements). Also, SPEED performs better with a larger number of iterations or simple material. A simplified upper limit of SPEED is found to be inversely proportional to the SOLVE ratio and the iterative ratio. Theoretically, the SPEED can go to infinity at negligible SOLVE ratio combined with zero iterative ratio. The analytical formula is extended to include the influence of the multi-incrementally updated super elements on the achieved SPEED. It is concluded that the three domain method accelerates the implicit method more than the two domain method for the same number of increments. The performance of the three domain algorithm has a similar response as the two domain algorithm regarding the iterative ratio and the SOLVE ratio. The analytical formula is validated and it can predict in advance the expected SPEED of an implicit simulation for localised deformation.

The task of the efficient implicit method for incremental forming is to control the update frequency of different domains. The size of the FE model has a direct impact on the partial cost of the solver and consequently on the performance of the method. The size of the system of equations can be reduced by including static condensation for the internal degree of freedoms of the pseudo-linear super elements. A demonstrative case study showed that the static condensation does not reduce the solver cost enough to compensate the cost of condensing the pseudo-linear element, resulting in a lower performance of SPEED compared to the achieved SPEED of the non-condensed approach. It is worth investigating the increase of condensation cost for large super element size that may be caused by the matrix multiplications or intricacies of the applied solver.

The size of the FE model can be kept as small as possible by adaptive remeshing. A study on adaptive remeshing concludes that intensive remeshing (refining and coarsening) results in less accurate results while refining only maintains the accuracy compared to the reference (initially fine) FE model and it accelerates the reference simulation by the same factor as the remeshing method. It is two times faster than the reference model. The efficient modelling (adaptive refinement) combined with the efficient use of the implicit time integration scheme (two domain) accelerates the standard implicit simulation for a small academic case study of the SPIF process by a factor of 3.6. Adaptive refinement accelerates the standard simulation twice and the two domain method accelerates the adaptive refinement simulation by a factor of 1.8. This implementation was tested on a single processer. For industrial application, it is recommended to use parallel computing in combination with the efficient implicit method and the adaptive refinement method.

Process mechanics

The continuous bending under tension (CBT) process is an incremental sheet forming process. The deformation mode is similar to the deformation that takes place in incremental sheet forming. Using the CBT process, it is shown that a stable high strain can be achieved by combined stretching and bending.

A 3-dimensional FE model is created, it is discretized by triangular shell elements based on discrete Kirchhoff theory. Three different meshes are investigated: coarse, intermediate and fine. The used different meshes show no significant difference in predicting the force–displacement curve of the process during the stable deformation. Early prediction of specimen failure is observed using the fine mesh discretization.

Three different material models are used: one isotropic hardening model and two isotropic/kinematic models. Despite the significant difference between these models in predicting the stress–strain curve after load reversal, only a slight difference between the models is observed in predicting the force–displacement curve in the CBT process. A major difference between the material models is observed in predicting the failure of the specimen. Combined with mesh dependency, it is concluded that the FE model cannot predict failure correctly.

A further analysis of the force–displacement curve for the CBT process is described based on an isotropic hardening material model. Focusing on the process description, the cyclic force–displacement curve consists of two parts: a steady part and a transient part (peak). The steady part of the cycle represents the deformation of the strip governed by significant curvature change of the strip because of bending. The peak results from the deformation of the strip by increasing the tension force with no significant change in strip curvature. The peak is observed after reversing the traveling direction of the roll set. During the process, the cycle pattern develops. This development is a result of two process factors: different zones of bending/unbending frequency and material migration through these zones. These main factors produce different levels of hardening and thickness through the length of the strip influencing implicitly the development of the cycle pattern.

A numerical stability criterion is derived for inhomogeneous stress distribution through the cross sectional area. The deformation is stable as long as a local extension requires an increase of the force. The model describes the importance of bending in stabilizing the deformation. It creates a compressive stress and elastically loaded fibers. The shift of a fiber from compression to tension increases the average stress. The presence of elastically loaded fibers maintains the average tangent stiffness at a high level as required by the stability criterion.

Several research groups have been interested in studying the mechanism(s) that govern(s) the SPIF process. Up to today, these mechanisms are studied individually and there is no general agreement on a specific mechanism. All proposed mechanisms are directly or indirectly related to bending. Very few researchers link the extended formability in SPIF to bending itself (based on FEM simulation by Sawada *et al.* (2001) and experimentally by Emmens (2006)). In this thesis, further development is carried out following the work of Hadoush *et al.* (2007) and Emmens and van den Boogaard (2009a) demonstrating how the bending mechanism does stabilize the deformation in incremental sheet forming to high strain.

Bibliography

Ambrogio, G., L. De Napoli, L. Filice, F. Gagliardi and M. Muzzupappa (2005a), Application of incremental forming process for high customized medical product manufacturing, *Journal of Materials Processing Technology*, vol. 162-163, pp. 156–162.

Ambrogio, G., L. Filice, F. Gagliardi and F. Micari (2005b), Sheet thinning prediction in single point incremental forming, *Advanced materials research*, vol. 6-8, pp. 479–486.

Bambach, M. (2008), *Process strategies and modelling approaches for asymmetric incremental sheet forming*, Ph.D. thesis, RWTH Aachen.

Bambach, M., J. Ames, M. Azaouzi, L. Compagne, G. Hirt and J. L. Batoz (2005), Initial experimental and numerical investigations into a class of new strategies for single point incremental sheet forming (SPIF), in: *The 8th International ESAFORM Conference on Material Forming*, pp. 671–674.

Bambach, M., B. Taleb Araghi and G. Hirt (2009), Strategies to improve the geometric accuracy in asymmetric single point incremental forming, *Production Engineering Research and Development*, vol. 3, pp. 145–156.

Batoz, J. L., K. J. Bathe and L. W. Ho (1980), Study of three–node triangular plate bending elements, *International Journal for Numerical Methods in Engineering*, vol. 15:12, pp. 1771–1812.

Batoz, J. L. and P. Lardeur (1989), A discrete shear triangular nine d.o.f. element for the analysis of thick to very thin plates, *International Journal for Numerical Methods in Engineering*, vol. 28:3, pp. 533–560.

Belytschko, T., W. K. Liu and B. Moran (2007), *Nonlinear finite elements for continua and structures*, John Wiley and Sons Ltd, England.

Belytschko, T., H. J. Yen and R. Mullen (1979), Mixed methods in time integration, *Computer Methods in Applied Mechanics and Engineering*, vol. 17-18, pp. 259–275.

Benedyk, J., D. Stawarz and N. Parikh (1971), A method for increasing elongation values for ferrous and nonferrous sheet metals, *J. of Materials*, vol. 6:1, pp. 16–29.

Bonet, J. (1994), Error estimators and enrichment procedures for the finite element analysis of thin sheet metal forming processes, *International Journal for Numerical Methods in Engineering*, vol. 34, pp. 1573–1591.

Brunssen, S. and B. Wohlmuth (2009), An overlapping domain decomposition method for the simulation of elastoplastic incremental forming processes., *International Journal for Numerical Methods in Engineering*, vol. 77, pp. 1224–1246.

Chaboche, J. (1991), On some modifications of kinematic hardening to improve the description of ratchetting effects, *International Journal of Plasticity*, vol. 7, pp. 661–678.

Cook, R. D., D. S. Malkus, M. E. Plesha and R. J. Witt (2002), *Concepts and application* *of finite element analysis*, John Wiley and Sons Ltd, England, 4th edn.

Dejardin, S., J. C. Gelin and S. Thibaud (2008), Finite element analysis and experimental investigation for improving knowledge in single point incremental sheet forming, in: *The 9th ICTP Incremental conference on Technology of Plasticity*, pp. 2002–2007.

Duflou, J. R., B. Lauwers, J. Verbert, Y. Tunckol and H. De Baerdemaker (2005), Achievable accuracy in single point incremental forming: case studies, in: *The 8th Esaform conference*, vol. 2, pp. 675–678.

Emmens, W. C. (2006), Water jet forming of steel beverage cans, *Int. J. of Machine Tools and Manufacture*, vol. 46:11, pp. 1243–1247.

Emmens, W. C., G. Sebastiani and A. H. van den Boogaard (2010), The technology of incremental sheet forming–a brief review of the history, *Journal of Materials Processing Technology*, doi:10.1016/j.jmatprotec.2010.02.014.

Emmens, W. C. and A. H. van den Boogaard (2009a), Incremental forming by continuous bending under tension – an experimental investigation, *J. of Materials Processing Technology*, vol. 209:14, pp. 5456–5463.

Emmens, W. C. and A. H. van den Boogaard (2009b), An overview of stabilizing deformation mechanisms in incremental sheet forming, *J. of Materials Processing Technology*, vol. 209:8, pp. 3688–3695.

Fratini, L., G. Ambrogio, R. Di Lorenzo, L. Filice and F. Micari (2004), The influence of mechanical properties of sheet material on formability in single point incremental forming, *Annals of CIRP*, vol. 53:1, pp. 207–210.

Hadoush, A. and A. H. van den Boogaard (2008), Time reduction in implicit single point incremental sheet forming simulation by refinement–derefinement, *International Journal of Material Forming*, vol. suppl 1, pp. 1167–1170.

Hadoush, A., A. H. van den Boogaard and J. Huétink (2007), Stable incremental deformation of a strip to high strain, *Key Engineering Materials*, vol. 344, pp. 615–620.

Henrard, C. (2008), *Numerical simulation of the single point incremental forming process*, Ph.D. thesis, University of Liege.

Hill, R. (1948), A theory of the yielding and plastic flow of anisotropic metals, in: *the Royal Society of London; Series A*, vol. 193, pp. 281–297.

Hill, R. (1950), *The mathematical theory of plasticity*, Clarendon Press, Oxford.

Hirt, G., J. Ames and M. Bambach (2005), A new forming strategy to realise parts designed for deep drawing by incremental CNC sheet forming, in: *Steel Research 71*, vol. 2/3, pp. 160–166.

Hirt, G., S. Junk and N. Witulski (2002), Incremental sheet forming: quality evaluation and process simulation, in: *The 7th ICTP International conference on Technology of Plasticity*.

Huerta, A., P. Díez and A. Rodriguez-Ferran (1998), Adaptivity and error estimation, in: *The 6th International Conference on Numerical Methods in Industrial Forming Processes*, pp. 63–74.

Hughes, T. J. R. and W. K. Liu (1978), Implicit-explicit finite elements in transient analysis: Implementation and numerical examples, *Journal of Applied Mechanics*, vol. 45, pp. 375–378.

Iseki, H., K. Kato and S. Sakamoto (1989), Flexible and incremental sheet metal forming using a spherical roller (in Japanese), in: *40th Japanese Joint Conference for the Technology of Plasticity*, pp. 41–44.

Jackson, K. and J. Allwood (2009), The mechanics of incremental sheet forming, *J. of Materials Processing Technology*, vol. 209, pp. 1158–1174.

Jeswiet, J. and E. Hagan (2001), Rapid proto-typing of a headlight with sheet metal, in: *The 9th International Conference on Sheet Metal*, pp. 165–170.

Jeswiet, J., F. Micari, G. Hirt, A. Bramley, J. Duflou and J. Allwood (2005), Asymmetric single point incremental forming of sheet metal, *Annals of the CIRP*, vol. 54:2, pp. 88–114.

Kim, S., Y. Lee, Y. Kwon and J. H. Lee (2008), Study on incremental formability of AZ31 alloy sheet, in: *Metal forming 2008, Steel Research International*, vol. 1, pp. 691–698.

Kreyszig, E. (1993), *Advanced Engineering Mathematics*, John Wiley and Sons Inc, New York, 7th edn.

Leszak, E. (1967), Apparatus and process for incremental dieless forming, *patent US3342051A1*.

Maidagan, E., J. Zettler, M. Bambach, P. P. Rodríguez and G. Hirt (2007), A new incremental sheet forming process based on a flexible supporting die system, *Key Engineering Materials*, vol. 44, pp. 607–614.

Marciniak, Z. and J. Duncan (1992), *The Mechanics of Sheet Metal Forming*, Edward Arnold, Great Britain.

Mason, B. (1978), Sheet metal forming for small batches, bachelor thesis, University of Nottingham.

Matsubara, S. (1994), Incremental backward bulge forming of a sheet metal with a hemispherical head tool (in Japanese), *Journal of the Japan Society for Technology of Plasticity*, vol. 35, pp. 1311–1316.

Meier, H., V. Smukala, O. Dewald and J. Zhang (2007), Two point incremental forming with two moving forming tools, *Key Engineering Materials*, vol. 44, pp. 599–605.

Meinders, T. (2000), *Developments in numerical simulations of real-life deep drawing process*, Ph.D. thesis, University of Twente.

Mols, A. (1972), Een mechanische techniek om het optreden van plastische on stabiliteit te voorkomen (in Dutch), *Internal report University of Delft*.

Przemieniecki, J. S. (1968), *Theory of matrix structural analysis*, McGraw–Hill Book Company, New York.

Quigley, E. and J. Monagan (2002), The finite element modelling of conventional spinning using multi–domain models, *Journal of Materials Processing Technology*, vol. 124, pp. 360–365.

Ramadan, M., L. Fourment and H. Digonnet (2009), A parallel two mesh method for speeding–up processes with localised deformation: application to cogging, in: *The 12th International ESAFORM Conference on Material Forming*.

Rojek, J., E. Oñate and E. Postek (1998), Application of explicit FE codes to simulation of sheet and bulk metal forming processes, *Journal of Materials Processing Technology*, vol. 80-81, pp. 620–627.

Sawada, T., G. Fukuhara and M. Sakanoto (2001), Deformation mechanism of sheet metal in stretch forming with computer numerical control machine tools (only the abstract is availabe in English), *J. of the Japan Society for Technology of Plasticity*, vol. 42:489, pp. 85–87.

Sebastiani, G., A. Brosius, A. E. Tekkaya, W. Homberg and M. Kleiner (2007), Decoupled simulation method for incremental sheet metal forming, in: *NUMIFORM '07, Materials*

Processing and design: modeling simulation and applications, pp. 1501–1506.

Simo, J. C. and T. J. R. Hughes (2000), *Computational inelasticity*, Springer, USA.

Smith, B. F., P. E. Bjorstad and W. Gropp (1996), *Domain decomposition*, Cambridge University Press, USA.

Van den Boogaard, A. H., T. Meinders and J. Huétink (2003), Efficient implicit finite element analysis of sheet forming processes, *International Journal for Numerical Methods in Engineering*, vol. 56:8, pp. 1083–1107.

Živković, M., M. Kojić, R. Slavković and N. Grujović (2001), A general beam finite element with deformable cross–section, *Computer Methods in Applied Mechanics and Engineering*, vol. 190, pp. 2651–2680.

Yoshida, F. and M. Urabe (1999), Fe simulation of tension levelling process for clad sheet metals, in: *the 6th ICTP*, pp. 1107–1112.

Zienkiewicz, O. C. and R. L. Taylor (2005), *The Finite Element Method: for Solid and Structural Mechanics*, Elsevier Butterworth, Barcelona, 6th edn.