

Substructuring in the implicit simulation of single point incremental sheet forming

The incrementally updated approach

A. Hadoush · A. H. van den Boogaard

Received: 23 August 2008 / Accepted: 2 March 2009
© The Author(s) 2009. This article is published with open access at Springerlink.com

Abstract This paper presents a direct substructuring method to reduce the computing time of implicit simulations of single point incremental forming (SPIF). Substructuring is used to divide the finite element (FE) mesh into several non-overlapping parts. Based on the hypothesis that plastic deformation is localized, the substructures are categorized into two groups: the plastic—nonlinear—substructures and the elastic—pseudo-linear—substructures. The plastic substructures assemble a part of the FE mesh that is in contact with the forming tool; they are iteratively updated respecting all nonlinearities. The elastic substructures model the elastic deformation of the rest of the FE mesh. For these substructures, the geometrical and the material behaviour are assumed linear within the increment. The stiffness matrices and the internal force vectors are calculated at the beginning of each increment then they are statically condensed to eliminate the internal degrees of freedom (DOF). In the iteration process the condensed stiffness matrices for the elastic substructures are kept constant. The condensed internal force vectors are updated by the multiplication of the condensed stiffness matrices and the displacement increments. After convergence, any geometrical and

material nonlinearity for the elastic substructures are nonlinearly updated. The categorization of substructures in plastic and elastic domains is adapted during the simulation to capture the tool motion. The resulting, plastic and condensed elastic, set of equations is solved on a single processor. In an example with 1600 shell elements, the presented substructuring of the SPIF implicit simulation is 2.4 times faster than the classical implicit simulation.

Keywords Substructuring · Incremental forming · Finite elements

Introduction

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 toolpath [1, 2], a sketch of SPIF is presented in Fig. 1. Because SPIF is a dieless process, it is perfectly suited for prototyping and small volume production. The simulation of SPIF results in enormous computing times for two reasons. First of all, the blank is deformed by a sequence of small increments that requires thousands of numerical increments to be performed. Secondly, the small contact area between the forming tool and the blank requires a very fine FE mesh to capture the introduced deformation. The extreme computing times currently limit the applicability of FE simulations to very simple academic samples.

The implicit simulation of SPIF provides a very good agreement with experimental data [3]. For nonlinear

A. Hadoush (✉)
Materials Innovation Institute, P.O. Box 5008,
2600 GA Delft, The Netherlands
e-mail: a.hadoush@m2i.nl

A. H. van den Boogaard
Faculty of Engineering Technology, University of Twente,
P.O. Box 217, 7500 AE Enschede, The Netherlands
e-mail: a.h.vandenboogaard@utwente.nl

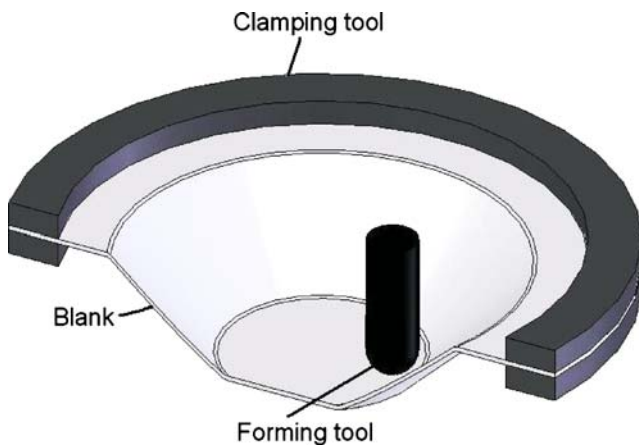


Fig. 1 Schematic representation of the SPIF process

implicit simulations, the Newton–Raphson method is often preferred because of its quadratic convergence behaviour although every iteration is relatively expensive. The modified Newton–Raphson method is cheaper per iteration, but it shows only linear convergence behaviour [4]. In SPIF simulation, it has been noticed that the overall computing time by a modified Newton–Raphson method is higher than the computing time by a full Newton–Raphson method, because it requires much more iterations per increment to converge.

The small sized tool 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 [5, 6]. The deformation in SPIF is modeled as localized plastic deformation [7]. According to this hypothesis plastic deformation is localized in a small zone in the region of the forming tool surrounded by elastic deformation of the rest of the blank. The small tool limits the numerical incremental step size, even though the implicit scheme is stable for much larger increment sizes. Using too large increments results in simulating a large number of penetrations instead of continuous incremental forming. In a simulation, the localized plastic deformation that is introduced by small increments results in a localized strong nonlinearity and a global weak nonlinearity. The strong nonlinearity is a combination of geometrical and material nonlinearity and it is located in the vicinity of the tool. The rest of the FE mesh will have a weak geometrical nonlinearity.

The basic idea, presented in this paper, is to treat the strong nonlinearity by a classical full Newton–Raphson iterative procedure, while the elastic deformation of the rest of the blank, the weak nonlinearity, is modeled

by linearized behaviour within the increment. After convergence of each increment, the geometrical and possibly small material nonlinearity are updated. This approach combines the advantage of a fast convergence for the highly nonlinear process in the plastic part with a cheap calculation in the much larger elastic part of the blank, significantly reducing the total computing time. Similar approaches have been introduced in computational mechanics and two of them are mentioned here. The first approach is the subcycling in explicit methods to overcome the problem of very small or very stiff elements [8]. The second approach is the implicit-explicit method, where part of the system Jacobian matrix is treated implicitly and part explicitly [9].

Domain decomposition was adopted early in structural analysis. The basic idea of domain decomposition is to divide the domain into subdomains. The subdomains are potentially solved by parallel computing. The subdomains sustain compatibility with each other by applying boundary conditions on the interfaces connecting the subdomains. Explicit overlap of the subdomains is possible. Non-overlapping domain decomposition is also known as the substructuring method [10]. Domain decomposition and substructuring are applied in nonlinear computational mechanics e.g. in large scale, millions of DOFs, elastic–plastic FE analysis of nuclear structures [11] and simulation of forming processes: extrusion [12], forging with deformable tools [13] and spinning [14].

The combined Newton–Raphson and condensed linearized elastic (CNRCLE) treatment of SPIF simulation is implemented by substructuring. The FE mesh is divided into plastic substructures and elastic substructures. The plastic substructures are treated by the full Newton–Raphson method while the elastic substructures are treated by the condensed linearized elastic approach. The elastic substructures internal DOFs, considered as slave DOFs, are statically condensed to its external (master) DOFs resulting in a significant reduction of the size of the system of equations (SOE). The reduced SOEs of the remaining master DOFs is formed by the assembly of the external DOFs of all substructures and the internal DOFs of the plastic substructures. The reduced SOEs is solved at once on a single processor. The time reduction of the CNRCLE calculation is obtained by the algorithm on a single processor, not necessarily by parallel computing. The CNRCLE algorithm can be modified to make use of parallel computing, but that is not the focus of this paper. The plastic-elastic substructure categorization is adapted during the simulation to capture the tool motion.

Substructuring

In this paper, direct substructuring is considered as a non-overlapping domain decomposition method. It is a way to organise the static condensation of large linear systems arising from the discretization of partial differential equations [10]. Briefly, consider a linear elastic FE mesh as shown in Fig. 2. The FE mesh is divided into 4 equal substructures. By substructuring, the initial SOEs is organised in 4 smaller SOEs. The partitioned SOEs of each substructure is

$$\begin{bmatrix} k_{ii} & k_{ie} \\ k_{ei} & k_{ee} \end{bmatrix}^s \begin{pmatrix} u_i \\ u_e \end{pmatrix}^s = \begin{pmatrix} f_i \\ f_e \end{pmatrix}^s \tag{1}$$

where u , k and f are the displacement vector, the stiffness matrix and the force vector. The subscripts i and e represent internal and external DOFs respectively. The superscript s refers to substructure number, $s = 1, 2, \dots, n$ where n is the total number of substructures in the FE model.

The internal (slave) DOFs are statically condensed (block factorized) to their external (master) DOFs as follows

$$u_i^s = (k_{ii}^s)^{-1} (f_i^s - k_{ie}^s u_e^s) \tag{2}$$

Such that the condensed form becomes

$$k_c^s u_e^s = f_c^s \tag{3}$$

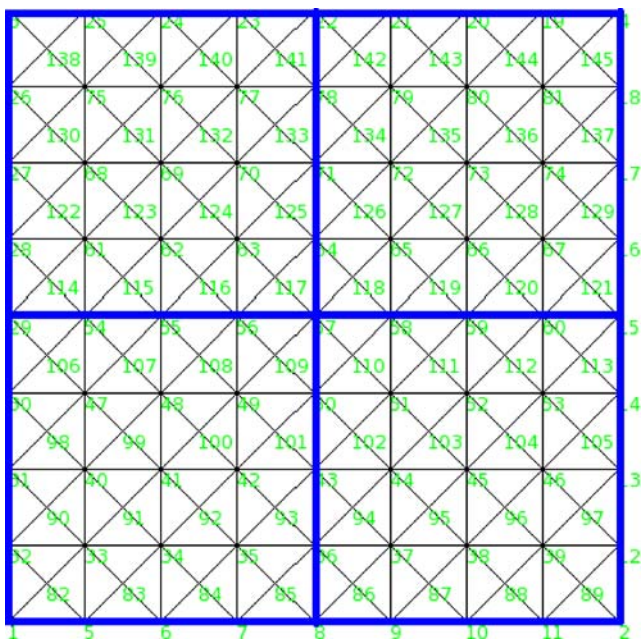


Fig. 2 Substructuring a FE mesh

where

$$k_c^s = k_{ee}^s - k_{ei}^s (k_{ii}^s)^{-1} k_{ie}^s \tag{4}$$

$$f_c^s = f_e^s - k_{ei}^s (k_{ii}^s)^{-1} f_i^s \tag{5}$$

k_c and f_c represent the condensed stiffness matrix and force vector, respectively. The condensed stiffness matrix in Eq. 4 represents what is known in substructuring as the Schur complement. The condensed SOEs of each substructure is presented in Eq. 3. A reduced global SOE is assembled from the condensed SOEs of the substructures. The reduced SOEs of the master DOFs is representative of the initial SOEs, but smaller and denser. The total number of unknown DOFs in the reduced SOEs are equal to the number of unknown DOFs at the substructure interfaces—the bold lines in Fig. 2. The reduced SOE is solved at once, then the internal DOFs of each substructure are evaluated by Eq. 2. It is important to mention that the condensation and the evaluation of the internal DOFs of each substructure can be done independently of the other substructures.

The CNRCLE

Now, a nonlinear SOEs is considered. Consider for example that an external force is applied at the lower left part of the FE mesh shown in Fig. 2. If the force is large enough it leads to a nonlinear response of the material and the geometry. By assumption, the nonlinearity is dominant in the lower left part of the FE mesh. A well known implicit approach to solve the nonlinear SOEs is the Newton–Raphson method [15]. In this section, the implicit solution of the nonlinear SOEs is substructured. First, the Newton–Raphson method is summarized, then the substructuring treatment is explained. The Newton–Raphson method updates an incremental displacement vector d with an iterative displacement vector Δd , using the tangent of the nonlinear SOEs $A(d)$ by solving

$$r(d) + A(d)\Delta d = 0 \tag{6}$$

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

$$r(d) = f_{\text{int}}(d) - f_{\text{ext}}(d) \tag{7}$$

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

$$A(d) = \frac{\partial r}{\partial d} = \frac{\partial f_{\text{int}}}{\partial d} - \frac{\partial f_{\text{ext}}}{\partial d} = k_{\text{int}} - k_{\text{ext}} \tag{8}$$

where k_{int} and k_{ext} are the tangent stiffness matrix and the load stiffness matrix, respectively. The linearized model is solved for the iterative change of the nodal displacements Δd

$$\Delta d = -A^{-1}r \tag{9}$$

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

$$d_{j+1} = d_j + \Delta d \tag{10}$$

where j is the iteration number. The new nodal displacements are checked for convergence. If convergence is not achieved, the linearized model is recalculated and solved for a new Δd .

The nonlinear SOEs is substructured based on a dominant nonlinearity in the lower left part of the FE mesh. The FE mesh is divided into 4 equal substructures. All the nonlinearity is located at the lower left substructure, substructure number 1. At the first iteration of each increment, the substructural residual r^s and the substructural effective tangent stiffness matrix A^s are calculated for all substructures separately:

$$r^s = \begin{pmatrix} r_i \\ r_e \end{pmatrix}^s = \begin{pmatrix} f_i \\ f_e \end{pmatrix}_{\text{int}}^s - \begin{pmatrix} f_i \\ f_e \end{pmatrix}_{\text{ext}}^s \tag{11}$$

$$A^s = \begin{bmatrix} A_{ii} & A_{ie} \\ A_{ei} & A_{ee} \end{bmatrix}^s = \begin{bmatrix} k_{ii} & k_{ie} \\ k_{ei} & k_{ee} \end{bmatrix}_{\text{int}}^s - \begin{bmatrix} k_{ii} & k_{ie} \\ k_{ei} & k_{ee} \end{bmatrix}_{\text{ext}}^s \tag{12}$$

The substructural linearized model becomes

$$\begin{pmatrix} r_i \\ r_e \end{pmatrix}_{j+1}^s = \begin{pmatrix} r_i \\ r_e \end{pmatrix}_j^s + \begin{bmatrix} A_{ii} & A_{ie} \\ A_{ei} & A_{ee} \end{bmatrix}_j^s \begin{pmatrix} \Delta d_i \\ \Delta d_e \end{pmatrix}^s \tag{13}$$

For substructures 2–4, r^s and A^s contain the internal force contribution only since there is no external force. The iterative displacements for the internal nodes Δd_i^s are expressed as function of the iterative displacements of the nodes on the boundary of the substructure Δd_e^s

$$\Delta d_i^s = \left(A_{ii,j}^s\right)^{-1} \left(-r_{i,j}^s - A_{ie,j}^s \Delta d_e^s\right) \tag{14}$$

The residual forces on the boundary nodes of substructure s can now be written in condensed form:

$$r_{c,j+1}^s = r_{c,j}^s + A_{c,j}^s \Delta d_e^s \tag{15}$$

where

$$A_{c,j}^s = A_{ee,j}^s - A_{ei,j}^s \left(A_{ii,j}^s\right)^{-1} A_{ie,j}^s \tag{16}$$

$$r_{c,j}^s = r_{e,j}^s - A_{ei,j}^s \left(A_{ii,j}^s\right)^{-1} r_{i,j}^s \tag{17}$$

The global condensed linearized model (GCLM) of the master DOFs is assembled of two contributions. The first contribution consists of r^s and A^s of the internal and the external DOFs for substructure 1. For this plastic substructure all DOFs are master DOFs. The second contribution consists of the condensed r_c^s and A_c^s of the external DOFs for substructures 2–4. The iterative displacement is solved by requiring $r_{j+1} = 0$ from the linearized model:

$$\begin{pmatrix} r_{mi} \\ r_{me} \end{pmatrix}_{j+1}^{\text{gclm}} = \begin{pmatrix} r_{mi} \\ r_{me} \end{pmatrix}_j^{\text{gclm}} + \begin{bmatrix} A_{mi-mi} & A_{mi-me} \\ A_{me-mi} & A_{me-me} \end{bmatrix}_j^{\text{gclm}} \times \begin{pmatrix} \Delta d_{mi} \\ \Delta d_{me} \end{pmatrix}^{\text{gclm}} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}^{\text{gclm}} \tag{18}$$

where the subscripts mi and me refer to the internal DOFs contribution of substructure 1 and the external DOFs contribution of all substructures, respectively. The size of the GCLM is significantly reduced by the condensation of the internal DOFs of substructures 2–4. The GCLM is solved for the unknown Δd^{gclm} at once. The total incremental nodal displacements for the master external and internal DOFs are updated by the addition of the corresponding iterative increment Δd .

$$d_{me,j+1} = d_{me,j} + \Delta d_{me} \tag{19}$$

$$d_{mi,j+1} = d_{mi,j} + \Delta d_{mi} \tag{20}$$

The residual of substructure 1 is recalculated using Eq. 11, while the condensed substructural residual r_c^s for substructures 2–4 are updated linearly using Eq. 15. 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

$$\psi = \frac{\|r\|}{\|r_{\text{int}}\|} \tag{21}$$

In general, the convergence is not achieved after the first iteration and more iterations are performed. The difference in treatment of the substructures becomes more clear after the first iteration. For substructure 1, r^s and A^s are recalculated by Eqs. 11 and 12. For substructures 2–4, the same value of A_c^s , that was calculated in the first iteration by Eq. 16 is used for the following iterations. The residual r_c^s of the first iteration is updated

linearly as in Eq. 15 using A_c^s of the first iteration. The condensed linear treatment of substructures 2–4 after the first iteration significantly reduces the computing time required for the rest of iterations. The reduction of the computing time is a result of the repeated use of A_c^s of the first iteration and the linear update of r_c^s .

Substructures 2–4 are treated linearly within an increment, but after convergence a final nonlinear evaluation is performed, based on the calculated displacements. In this evaluation, small non-linearities in the material response and geometrical non-linearities are fully considered, to a large extent correcting small errors due to the linearization.

Categorization

Considering SPIF, according to the localized plastic deformation hypothesis the plastic deformation is localized in a small zone in the region of the forming tool surrounded by elastic deformation of the rest of the blank. For that reason, the tool location is used to categorize the substructured FE mesh into plastic substructures and elastic substructures. The plastic substructures contain the major localized plastic deformation while the elastic substructures contain the elastic deformation and any small non-localized plastic

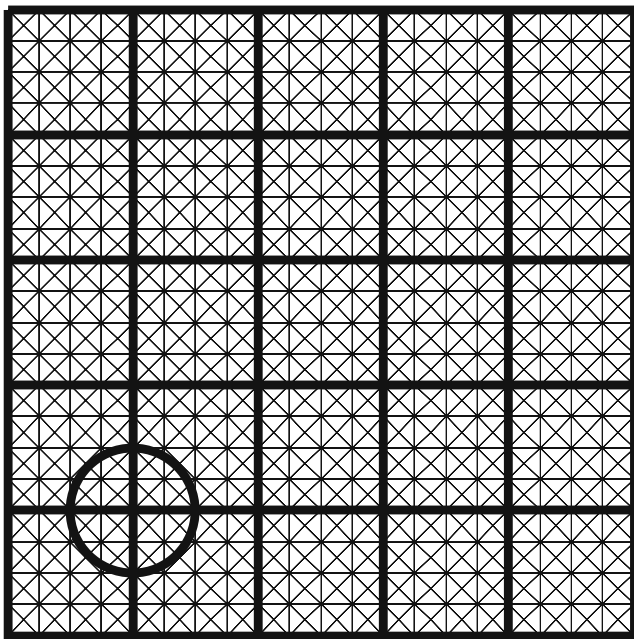


Fig. 3 Substructuring 1600 triangular element into 25 substructures, the 4 substructures in the vicinity of the tool (the circle) are plastic substructures and the others are elastic substructures

deformation if it occurs. A number of virtual cross points are introduced in the FE mesh. The cross point is a common node between 4 adjacent substructures. In Fig. 3 the substructures are visualized with thick lines. Each substructure is attached to at least one cross point. The closest cross point to the tool center categorizes the adjacent substructures as plastic substructures while the rest of the substructures are categorized as elastic substructures. The movement of the tool changes the active cross point and as result the elastic–plastic substructures distribution.

Implementation

The CNRCLE method is implemented in the in-house implicit finite element code DIEKA. Two tests are introduced to investigate the accuracy of the CNRCLE method. The first test focuses on the treatment of geometrical nonlinearity in an elastic substructure. The second test investigates the loading–unloading for a FE mesh that is categorized into plastic and elastic substructures.

Influence of geometrical nonlinearity

The focus of this test is to investigate the influence of the linearization in an elastic substructure with respect to geometrical nonlinearities. For this purpose, an initially flat strip of $50 \times 5 \times 1.2$ mm with elastic material is modelled with 160 shell elements as plotted in Fig. 4. The edge on the left of the strip is completely clamped. The edge on the right is moved 5 mm upwards while the rotation is suppressed. Four increment sizes are considered 0.01, 0.1, 0.2 and 0.5 mm.

The finite elements of the strip are grouped in one substructure. Two simulations are carried out per increment size. In the first simulation, the geometrical nonlinearity is treated iteratively like in the classical implicit method (Iterative). In the second simulation the substructure is selected as elastic. The geometrical nonlinearity is neglected within the increment and updated nonlinearly at the end of the increment as explained in the CNRCLE method (Increment).

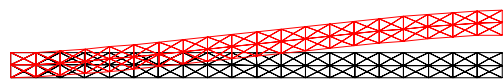


Fig. 4 The initial FE mesh of a strip (bottom) and the FE mesh at the end of the deformation (top)

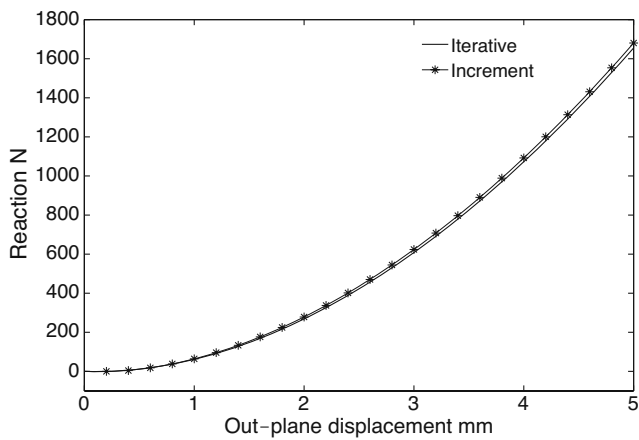


Fig. 5 A comparison between the reaction force achieved by the incremental update and the classical implicit approach (iterative) for an increment size of 0.2 mm

To compare the results, the in-plane reaction force (longitudinal) at the left edge for a displacement increment of 0.2 mm is plotted in Fig. 5. The achieved reaction force by the incremental approach has a very good agreement with the classical implicit (iterative) approach with a maximum error of 1.7%. The achieved reaction by the incremental approach for the other increment sizes has the same pattern, the error is plotted in Fig. 6. As expected, the largest increment size of 0.5 mm gives the largest error of 4%. In ISF simulations the displacement per increment will usually be much smaller. Under these conditions, the incrementally linearized approach can be considered valid.

Plastic loading and elastic unloading of a blank

In this test, the accuracy of the CNRCLE algorithm is investigated in the simulation of tool penetration and

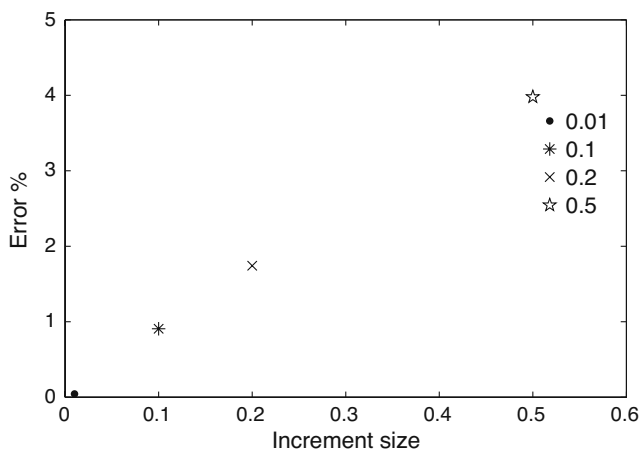


Fig. 6 The error in reaction force introduced by the incremental compared to the classical implicit approach

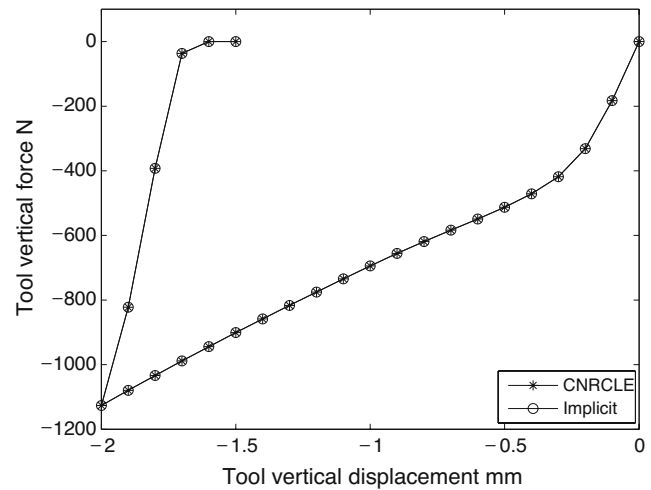


Fig. 7 Tool forces for the CNRCLE and the classical implicit approach in a penetration test

retraction on a clamped plate. This is representative for the first and last stage of an ISF process. In the loading stage the plate is deforming plastically near the tool. In the unloading stage, the plate shows elastic spring back.

A 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 respectively. The FE mesh and position of the tool are shown in Fig. 3. The 4 substructures in the vicinity of the tool are plastic substructures while the rest of the substructures are elastic during the entire simulation. The simulation is performed by the CNRCLE method and compared to the classical implicit method without substructuring. To compare the results, the calculated vertical force on the tool is plotted in Fig. 7. The result of the CNRCLE coincides with the classical implicit result.

Case study

To investigate the performance of the CNRCLE approach in speeding up the classical implicit simulation of SPIF, the production of a 45° pyramidal shape is simulated with both methods. The 20 mm deep pyramid is made out of a 100 × 100 × 1.2 mm initially flat blank. An analytical spherical tool of 10 mm radius is used. The tool follows a counter clockwise tool path for 40 loops. At the start of 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 40. The blank edges are completely suppressed for the entire

simulation. Each simulation is performed on one core of a Sun Fire X4450 server with Intel Xeon X5365 3 GHz processors.

FE model

The numerical blank is discretized with 1600 triangular shell elements. The element type is the discrete shear triangle DST for bending, combined with a linear membrane element. The element has 6 DOFs per node. It has 3 integration points in plane and 5 in thickness direction (total 15). For the CNMCLE simulation, the elements are grouped in 100 substructures with 81 virtual cross points to categorize the elastic–plastic substructure distribution. In order to focus on the substructuring technique only, the material model is kept as simple as possible. The isotropic yield behaviour of the material is modelled with the von Mises criterion. The work hardening is governed by the Swift relation:

$$\sigma = 500(\varepsilon + 0.00243)^{0.2} \quad (22)$$

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 realistic calculation, the authors acknowledge that a better material model is required, that includes the anisotropic behavior of the sheet and the cyclic mode of deformation.

Results and discussion

The CNRCLE method is 2.4 times faster than the classical implicit method. With CNRCLE, the simulation

Table 1 Performance in SPIF simulation

	Classical	CNRCLE
CPU (hours)	6.73	2.82
Number of increments	9064	9081
Incremental cost (seconds)	2.67	1.12

requires 2.82 hours to finish while it requires 6.73 h with the classical implicit method, the performance of both simulations are summarized in Table 1. A relative unbalance force convergence criterion of 0.01 is used for both simulations. The average cost of each iteration in the classical implicit method is 0.6 s. For each iteration in the classical implicit method, the major consumption of the computing time is done by the calculation of r and A and performing the stress update (50% and 28.6%, respectively). A direct solver is used to solve the SOEs. For this particular case study the consumed computing time by the direct solver is not significant (5.3%) compared to the major consumptions. Each increment requires on average 3 iterations to converge.

The computing time for calculating r and A and performing the stress update scale almost linearly with the number of elements in the model. This aspect significantly contributes to the reduction in computing time with the CNRCLE method. For the first iteration in CNRCLE, the calculation of r and A requires the same computing time as the classical implicit method, since a nonlinear treatment is considered for all substructures. Additional computing time is required in the first iteration to perform the condensation. The condensation requires 10% of the cost of an iteration in the classical implicit approach. The stress update is performed only for the plastic substructures, reducing the required computing time by a factor RPT , where RPT is the ratio of the number of elements in the

Fig. 8 Vertical displacement for the classical implicit method (*left*) and the CNRCLE method (*right*)

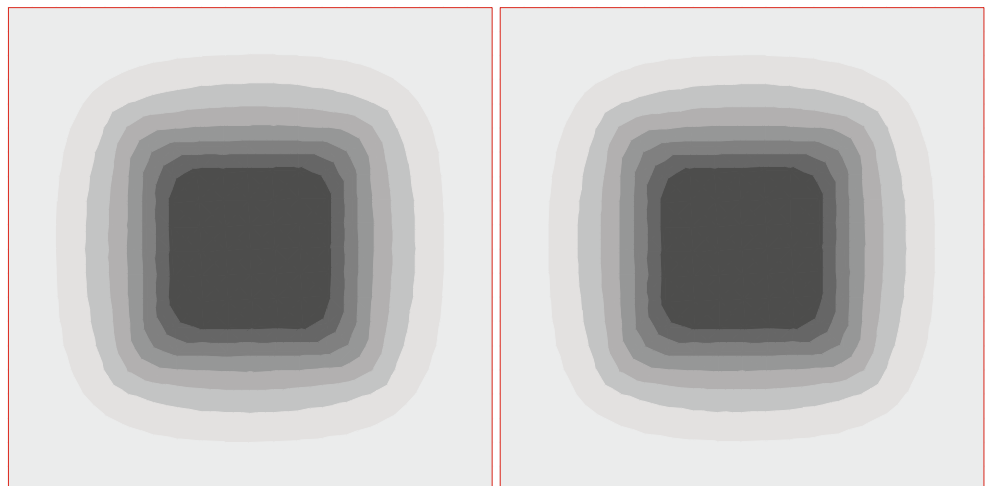
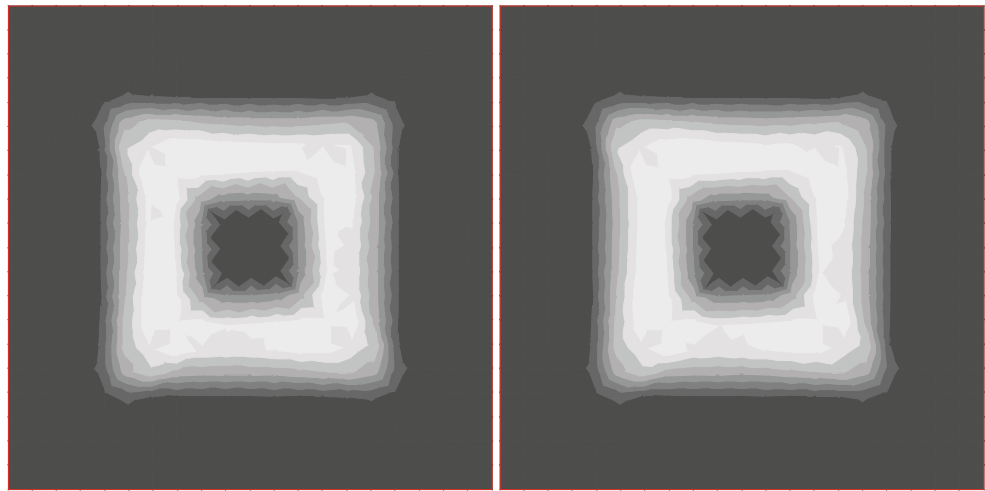


Fig. 9 Equivalent plastic strain at the highest integration point for the classical implicit method (*left*) and the CNRCLE method (*right*)



plastic substructures to the total number of elements in the model. In this case study, the number of element in the plastic substructures is 64 and the total number of element is 1600 making $RPT = 0.04$. The computing time for the linear update of r_c^s and the evaluation of d_i^s is small compared to the calculation of the plastic part because it is just a matrix–vector multiplication. Each iteration after the first iteration equals the iteration in classical implicit method scaled roughly by RPT . At the end of the increment, the stress is updated fully nonlinearly for the elastic substructures. The increment in CNRCLE method requires on average 2 iterations to converge.

The condensation of the internal DOFs of the elastic substructures is currently done by eliminating one DOF at the time. The condensation cost becomes expensive for larger substructures, e.g. the condensation of a system that has twice the number of DOFs increases by a factor 8. Of course, an expensive computing time for the condensation reduces significantly the reduction of the overall computing time. To maintain a significant reduction in the computing time for larger substructures, a more efficient condensation algorithm has to be implemented. A further reduction in computing time can be obtained by using the same condensed A_c^s and r_c^s for the elastic substructures for several increments. These options are subjects of ongoing research.

In Figs. 8 and 9 the displacements and equivalent plastic strains, obtained with the classical approach and with the CNRCLE approach are presented. The results show a very good agreement. The vertical displacement is plotted in Fig. 8 and shows a maximum value of 19.95 mm for the classical implicit method. The CNRCLE method achieved almost the same contour distribution and less than 0.05% error for the maximum

value. The equivalent plastic strain for the highest plane of integration points is plotted in Fig. 9, the maximum achieved value is 0.855 for both methods. The contour distribution for the CNRCLE is almost equal to the contour distribution for the classical implicit method.

Conclusions

A nonlinear analysis method with substructuring was presented for situations where strong local nonlinearities are combined with only weak nonlinearities in a large part of a model. The localized strongly nonlinear zone may move over the model during the simulation. This is typically the case in Single Point Incremental Sheet Forming (SPIF).

In the presented case study for simulation of SPIF, The CNRCLE method speeds up the implicit simulation 2.4 times. A very good agreement of results achieved with the CNRCLE method is found, in comparison with the results achieved with the classical implicit method. Categorization of the substructures into plastic substructures and elastic substructures fits the SPIF nature of forming. In more complex SPIF processes, where the deformation is more localized, the efficiency of the discussed method can be expected to be even higher.

Acknowledgements This research was carried out under the project number MC1.05227 in the framework of the Research Program of the Materials innovation institute M2i (www.m2i.nl), the former Netherlands Institute for Metals Research.

Open Access This article is distributed under the terms of the Creative Commons Attribution Noncommercial License which permits any noncommercial use, distribution, and reproduction in any medium, provided the original author(s) and source are credited.

References

1. Leszak E (1967) Apparatus and process for incremental dieless forming. Patent US3342051A1
2. Matsubara S (1994) Incremental backward bulge forming of a sheet metal with a hemispherical tool. *J Jpn Soc Technol Plast* 35:1311–1316
3. Bambach M, Hirt G, Ames J (2004) Modelling of optimization strategies in the incremental sheet metal forming process. NUMIFORM, Columbus, Ohio
4. Zienkiewicz OC, Taylor RL (2005) *The finite element method for solid and structural mechanics*. Elsevier Butterworth Heinemann, Oxford, pp 48–51
5. Jeswiet J (2005) Asymmetric incremental sheet forming. *Adv Mater Res* 6–8:35–58
6. Emmens WC, van den Boogaard AH (2008) An overview of stabilizing deformation mechanisms in incremental sheet forming. *J Mater Process Technol*. doi:[10.1016/j.jmatprotec.2008.10.003](https://doi.org/10.1016/j.jmatprotec.2008.10.003)
7. Hirt G, Junk S, Witulski N (2002) Incremental sheet forming: quality evaluation and process simulation. 7th ICTP Incremental conference on Technology of Plasticity. Yokohama, Japan
8. Belytschko T, Yen HJ, Mullen R (1979) Mixed methods in time integration. *Comput Methods Appl Mech Eng* 17–18: 259–275
9. Hughes TJR, Liu WK (1978) Implicit-explicit finite elements in transient analysis: implementation and numerical examples. *J Appl Mech* 45:375–378
10. Smith BF, Bjorstad PE, Gropp WD (1996) *Domain decomposition*. Cambridge University Press, Cambridge, pp 101–107
11. Miyamura T, Noguchi H, Shioya R, Yoshimura S, Yagawa G (2002) Elastic–plastic analysis of nuclear structures with millions of DOFs using the hierarchical domain decomposition method. *Nucl Eng Des* 212:335–355
12. Park K, Yang DY (2000) Mismatching refinement with domain decomposition for the analysis of steady state metal forming process. *Int J Numer Methods Eng* 48:1089–1106
13. Qingchun Y, Shengnian Q, Qi Z, Naiyan M (1997) FE simulation method for a forging system. *J Mater Process Technol* 63:678–683
14. Quigley E, Monagan J (2002) The finite element modelling of conventional spinning using multi domain models. *J Mater Process Technol* 124:360–365
15. Belytschko T, Liu WK, Moran B (2007) *Nonlinear finite elements for continua and structures*. Wiley, London, pp 319–325