An Adaptive Thick Shell Element for Crashworthiness Assessment of Laminated Composites

Johannes Främby[a], Jesper Karlsson[b], Martin Fagerström[a]
[a] Dept. Industrial and Materials Science, Div. Material and Computational Mechanics, Chalmers University of Technology, Gothenburg, Sweden
[b] DYNAmore Nordic AB, Gothenburg, Sweden

Abstract

The automotive industry is strongly dependent on efficient numerical tools in order to assess the crashworthiness of laminated composites. Unfortunately, to achieve predictive assessments of fracture in laminated composites one must resort to computationally costly, high-fidelity layered models, which in practice makes full vehicle crash simulations very difficult (or even impossible). One solution to this is to use an adaptive modelling technique where an initially coarse model is automatically refined, when and where needed, during the analysis.

In this context, we have developed an LS-DYNA® user element which can be adaptively refined through the thickness to allow for both so-called weak discontinuities (discontinuities in strain at material interfaces) and strong discontinuities (discontinuities in displacements, i.e. delamination cracks). Furthermore, we have proposed a remedy to the numerical instabilities which arise from using adaptive refinement in a dynamic explicit solver. This adaptive element proves capable of reproducing the result of high-fidelity models, although at a lower computational cost.

Introduction

The introduction of laminated Fibre-Reinforced Polymers FRP in the automotive industry is strongly dependent on accurate and efficient modelling tools to predict the correct energy absorption in crash simulations [1].

The numerical approaches to modelling laminated FRP in crash simulations can generally be divided into two categories where the focus is on either efficiency or accuracy:

1. With focus on efficiency, Equivalent Single-Layer (ESL) models with one element through the thickness are used together with phenomenological material models. The drawback is that it requires extensive physical testing on the laminate level and the response of a laminate with an arbitrary layup cannot be predicted.

2. With focus on accuracy, high-fidelity LayerWise (LW) models are adopted. Here, each individual ply is represented by a layer of separate elements. To account for delaminations, the interfaces are modelled by means of an interface cohesive zone (CZ) law in the form of interface elements or similar. This approach is generally able to accurately predict the failure process in laminated composites. However, the high resolution leads to extreme computational times, making industrial crash simulations infeasible.

A solution to reduce computational cost while maintaining the same level of accuracy as high-fidelity models is to resort to an adaptive modelling technique, where an initially coarse model can be locally refined during the simulation. The costs associated with LW models can then be limited to the areas where it is needed. To facilitate adaptive modelling in industrial crash simulations, a modified version of our previously presented adaptive method [2], has therefore been implemented as a user element in LS-DYNA.

In this paper, we will briefly describe the numerical implementation of the proposed adaptive method, including remedies to the instabilities that can occur when refining elements during a simulation in an explicit solver. For more details we refer to two up-coming journal articles [3] and [4].

Implementation of adaptive refinements

In summary, the adaptive method consists of the following steps:

1. The laminated structure is initially represented by a single layer of solid shell elements through the thickness.
2. Refinement indicators are used to locate areas which need to be refined.
3. The shell elements are refined through the thickness by enriching the element kinematics to account for material interfaces (weak discontinuities).
4. At interfaces prone to delaminate, CZ elements are inserted such that delaminations (strong discontinuities) can initiate and propagate.

In the first refinement stage, weak discontinuities (discontinuities in strain) are introduced through the thickness, which will transform the ESL element into a LW element. This way, better prediction of the out-of-plane stresses is achieved, and failure of individual layers can be modelled more accurately. However, initiation and propagation of delaminations are not possible and if this is needed there is a second refinement stage. In this stage, the weak discontinuities are transformed to strong discontinuities (discontinuities in displacements), and CZ elements are inserted. In the following, the first refinement step will be referred to as *weak refinements* and the secondary refinement step, will be referred to as *strong refinements*. In summary, every adaptive element in the model can be in one of three stages as illustrated in Figure 2: a) Unrefined; b) Weakly refined in one or several material interfaces or c) Weakly or strongly refined in one or several interfaces.

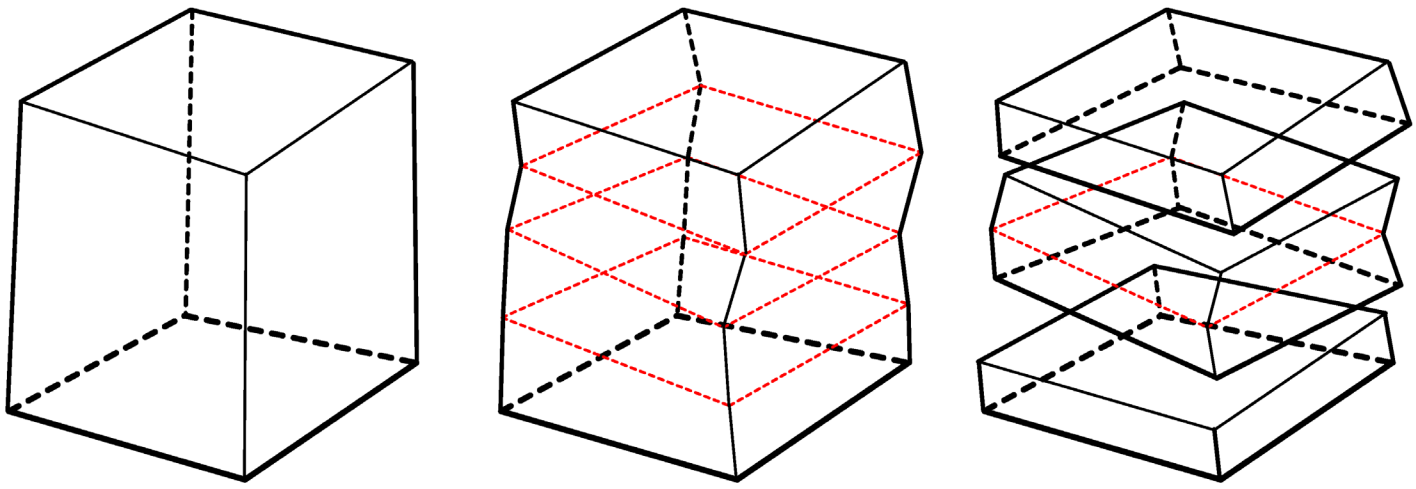

Figure 1 Different refinement stages of the adaptive elements. (a): In the unrefined stage the entire laminate is represented by one element through the thickness. (b): The element can be through-the-thickness refined in order to model weak discontinuities, here exemplified with three refinements. (c): The refined interfaces from the previous stage can be transformed to strong discontinuities (including CZ elements).

In order to limit the computational costs, the proposed method only allows through-the-thickness refinements, i.e. no in-plane refinements are made. Thus, while interlaminar fracture can be explicitly modelled (with strong refinements), intralaminar cannot. Instead, it is assumed that the intralaminar behaviour is modelled using a ply material model, e.g. the model by Costa [5]. However, the proposed adaptive method is not dependent on the type of intralaminar material model.

Adaptive element kinematics

Our implementation is based on a solid shell element (eight nodes with only translational Degrees of Freedom, DOF) where the refinements are represented using internal subelements, i.e. using an Augmented FEM [6] approach. This is achieved by utilising extra nodes with three translational DOF each.

Remark: Currently LS-DYNA does not allow for the extra nodes to be dynamically allocated, i.e. all (including the non-active) extra nodes are updated by the time integration during the entire simulation. Thus, any improved computational efficiency is currently only related to the internal computations of the element.

Element connectivity and internal force calculation

The base element consists of 8 vertex nodes. These base nodes are each associated with N extra nodes, making a total of $8(N+1)$ elemental nodes, where N is the number of active refinements in the element. Please note that the extra nodes share the same connectivity as their parent base nodes. Furthermore, we want to emphasise that unless the analysis input specifies otherwise, no refinements are present from the start. These are activated from within the elements when certain conditions are met.

The extra nodes represent the surfaces of the internal subelements and their reference positions $\boldsymbol{X}$ are an interpolation of the bottom and top base nodes as

$$\begin{aligned}
\boldsymbol{X}_{8(i-1)+5} &= \boldsymbol{X}_{8i+1} = \widehat{\boldsymbol{X}}_1 + \alpha_i\left(\widehat{\boldsymbol{X}}_5 - \widehat{\boldsymbol{X}}_1\right) \\
\boldsymbol{X}_{8(i-1)+6} &= \boldsymbol{X}_{8i+2} = \widehat{\boldsymbol{X}}_2 + \alpha_i\left(\widehat{\boldsymbol{X}}_6 - \widehat{\boldsymbol{X}}_2\right) \\
\boldsymbol{X}_{8(i-1)+7} &= \boldsymbol{X}_{8i+3} = \widehat{\boldsymbol{X}}_3 + \alpha_i\left(\widehat{\boldsymbol{X}}_7 - \widehat{\boldsymbol{X}}_3\right) \\
\boldsymbol{X}_{8(i-1)+8} &= \boldsymbol{X}_{8i+4} = \widehat{\boldsymbol{X}}_4 + \alpha_i\left(\widehat{\boldsymbol{X}}_8 - \widehat{\boldsymbol{X}}_4\right)
\end{aligned} \quad i = 1\ldots N, \tag{1}$$

where $\widehat{\boldsymbol{X}}$ are the reference positions of the base nodes and $|\alpha_i| \in]0,1[$ is a parameter which specifies the relative position of the interface with respect to the bottom and top surfaces. In Eq. (1) we assume that each internal interface consists of two coinciding surfaces, defined by eight extra nodes. However, during the first weak refinement stage only one set of four extra nodes is active while an additional set of four is reserved for later use. If the interface is then strongly refined the reserved set of nodes is activated.

Let $\boldsymbol{x}_k$ be the current coordinates of node k. Then

$$\boldsymbol{x}_k = \boldsymbol{X}_k + \boldsymbol{u}_k \qquad k = 1\ldots 8(N+1), \tag{2}$$

where $\boldsymbol{u}_k$ is the displacement of the node. The displacements of the extra nodes are calculated by the normal LS-DYNA time integration – meaning that forces (and masses) to all the extra DOF must be assembled by the user-element routine, which will be described next.

In each time step the user element is fed with the reference and current nodal coordinates $\boldsymbol{X}$ and $\boldsymbol{x}$ as well as the velocities $\boldsymbol{v}$. In return, the user element gives the element internal force vector $\boldsymbol{f}$. In the third refinement stage some of the interfaces can be weak refinements while other represent strong discontinuities with CZ elements (strong refinements). This means that the full internal force vector $\boldsymbol{f}$ of the element will be a sum of the forces from the currently active internal sublaminate and CZ elements, $\boldsymbol{f}^{\text{lam}}$ and $\boldsymbol{f}^{\text{coh}}$ respectively:

$$\boldsymbol{f} = \boldsymbol{f}^{\text{lam}} - \boldsymbol{f}^{\text{coh}}. \tag{3}$$

We will in the following refer to the sublaminate elements only as *subelements*.

It is important to note that that the through-the-thickness integration scheme does not change during refinements. Each ply is represented by one layer of integration points (IP). When a refinement is activated, the ply IP below the refinement are assigned to a lower subelement and those above to an upper subelement. That is, no mapping of IP data is required. This subdivision of the ply IP is exemplified in Figure 3. Thus, the main task for calculating Eq. (3) is to gather the correct nodal positions and velocities and ply data for each subelement and then assemble the force contributions to the right places in $\boldsymbol{f}$.

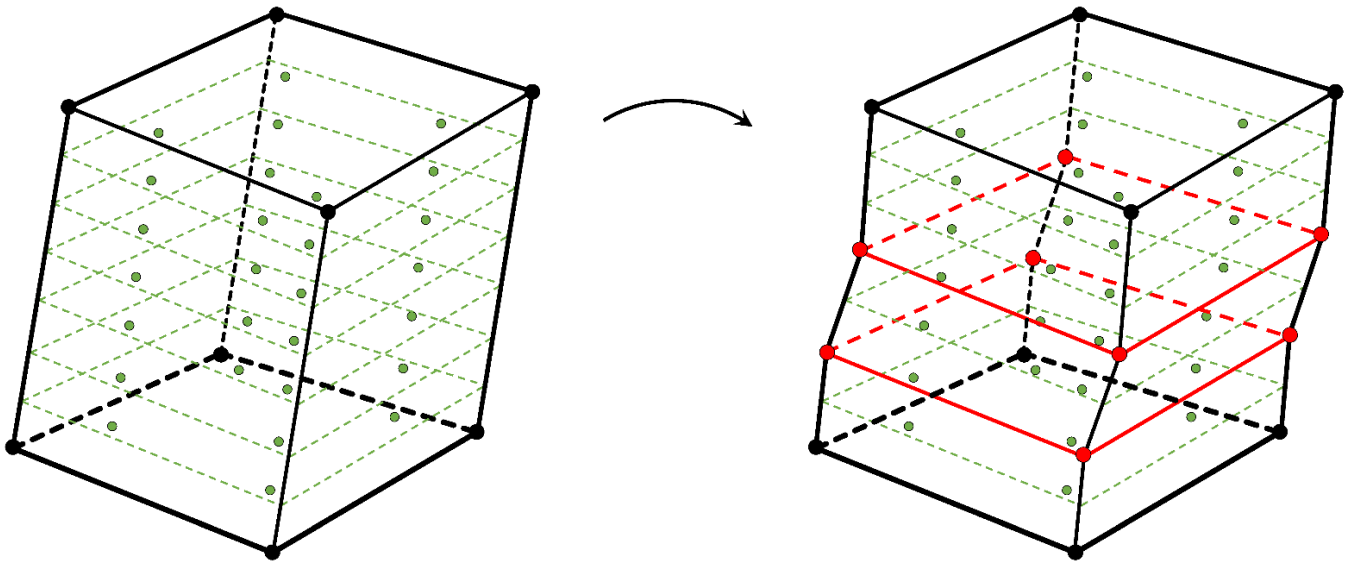

Figure 2. Example of how an unrefined element with seven plies, with one layer of IP each (left), is subdivided into subelements when the element is refined in interface two and four (right). The bottom two plies will be associated with the bottom subelement, plies three and four with the second subelement and the top three with a third subelement.

From the parent element $\boldsymbol{X}$, $\boldsymbol{x}$ and $\boldsymbol{v}$ the corresponding subvectors associated with subelement i can be extracted using projection matrices $\boldsymbol{\mathcal{P}}$ as

$$\boldsymbol{X}^{\text{sublam},i} = \boldsymbol{\mathcal{P}}^{\text{sublam},i}\boldsymbol{X}, \quad \boldsymbol{x}^{\text{sublam},i} = \boldsymbol{\mathcal{P}}^{\text{sublam},i}\boldsymbol{x}, \quad \boldsymbol{v}^{\text{sublam},i} = \boldsymbol{\mathcal{P}}^{\text{sublam},i}\boldsymbol{v}. \tag{4}$$

The position and velocity vectors in Eq. (4) together with the sublaminate material properties and IP data are used to call an element subroutine which calculates the subelement internal force vector $\boldsymbol{f}^{\text{sublam},i}$ of the current time step. In our current implementation the element subroutine is basically the same as the *PART_COMPOSITE_TSHELL, but with the assumption of equally thick layers of the same material.

Once the sublaminate forces are calculated, they are assembled in the element force vector using the projection matrices above as

$$\boldsymbol{f}^{\text{lam}} = \sum_{i=1}^{N+1} \boldsymbol{\mathcal{P}}^{\text{sublam},i^T} \boldsymbol{f}^{\text{sublam},i}. \tag{5}$$

For the strong refinement, the nodal positions and velocities associated with the CZ element can be extracted in a similar way as for the subelements. That is, for an interface i

$$\boldsymbol{X}^{\text{coh},i} = \boldsymbol{\mathcal{P}}^{\text{coh},i}\boldsymbol{X}, \quad \boldsymbol{x}^{\text{coh},i} = \boldsymbol{\mathcal{P}}^{\text{coh},i}\boldsymbol{x}, \quad \boldsymbol{v}^{\text{coh},i} = \boldsymbol{\mathcal{P}}^{\text{coh},i}\boldsymbol{v}, \tag{6}$$

where, once again, $\boldsymbol{\mathcal{P}}^{\text{coh},i}$ are projection matrices.

Similar to the subelements, the vectors in Eq. (6) together with the interface material properties are used to call a CZ element subroutine which calculates the internal force vector $\boldsymbol{f}^{\text{coh},i}$ of the current time step. Specifically, the subroutine is the same as solid element formulation 19 using material model 138 *MAT_COHESIVE_MIXED_MODE. When the cohesive force is obtained it is assembled to the element force vector as

$$\boldsymbol{f}^{\text{coh}} = \sum_{i \in I} \boldsymbol{\mathcal{P}}^{\text{sublam},i^T} \boldsymbol{f}^{\text{sublam},i}, \tag{7}$$

where I is an array containing those interfaces which have CZ element.

Adaptivity

Weak refinements is introduced by monitoring the intralaminar stress state with respect to the failure initiation level.

In order to predict delaminations correctly, the primary sources for these must be captured: high out-of-plane stresses and intralaminar cracks. The former can be considered by evaluating a cohesive failure initiation criterion at the interface positions. This requires good prediction of the out-of-plane stresses, which is of low quality in the unrefined shell. In [2] we showed a stress recovery technique to improve such a prediction . Furthermore, delaminations driven by intralaminar cracks can be captured by monitoring the intralaminar damage state. If a ply crack is beginning to form, strong refinements should be introduced adjacent to the crack. Note that other error indicators could be added depending on which phenomena that should be captured.

When a weak refinement is activated, its reference nodal positions $\boldsymbol{X}$ will be given by Eq. (1), but the current positions $\boldsymbol{x}$ and velocities $\boldsymbol{v}$ are unknown. In our model, we assign these in a similar way as the reference positions, i.e. by interpolating from the closest active nodes below and above.

When a strong refinement is activated, the weak refinement interface is duplicated by copying the current positions $\boldsymbol{x}$ and velocities $\boldsymbol{v}$ from the already activated set of extra nodes to the reserved set. The interface is now defined by eight nodes and a CZ element can be inserted in order to model delamination initiation and propagation.

Numerical stabilisation

Both the weak and the strong refinements will result in a sudden change in the internal forces, which may cause non-physical oscillations in the model. To minimise these, we have implemented two types of damping procedures.

During weak refinement, one source of the non-physical oscillations is that the interpolated positions (and velocities) of the newly activated extra nodes are likely not correct. So, in order to stabilise the refinement, a small amount of damping is applied for a short time period directly after the refinement.

To stabilize the strong refinement we follow a method similar to that by Menouillard and Belytschko [7], where a correction force is applied to balance the sudden change in forces. The purpose of the correction force is to tie the node pairs of the two coincidental interfaces together. The introduction of the strong refinements is made by gradually removing the tied force over a short time period.

Nodal mass

For the time integration to be performed, the correct nodal masses must be defined. During the initiation phase LS-DYNA will assemble mass to the base nodes by summing the contribution from all connected elements. Since the extra nodes are not active during the initiation phase, they are not assigned any mass. Therefore, if a refinement is activated the mass of both the base and extra nodes must be updated to accurately represent their associated subelements. Each time a refinement is activated, the old unrefined (sub)element mass is subtracted from the old nodes followed by adding the new subelements masses to the old nodes and the newly activated extra nodes.

Time step

The initial critical time step for the unrefined element can be calculated by LS-DYNA. However, when the element is refined there is the risk of violating the critical time step due to decreased (sub)element sizes and the addition of CZ elements. To avoid this, we have implemented an internal time step calculation that adjusts the computational time step appropriately.

Numerical examples

In this section we present two numerical examples to verify that we can recover the same results as a reference model (standard LS-DYNA element) and reproduce experimental results. In both examples solid shell element formulation 3 is used. For choosing the parameters which control the refinements and the stabilisation thereof we refer to [3] and [4].

Triple cantilever beam

In a first example we simulate a triple cantilever beam (TCB) where two mode I delaminations initiate and propagate through the beam. The beam consists of eight plies (with fibres along the beam) of a prototype material with properties given in Table 1. Since there is no pre-crack present, we manually decide that refinements should be made in the second, fourth and sixth interface. The second and sixth can delaminate and the fourth is a material interface with a thickness symmetry condition (when active) to force two delaminations. The beam is loaded down- and upwards in one end and is clamped in the other end, see Figure 4. An in-plane element size of 0.5 mm

is chosen and a prototype CZ element material with propertied to achieve stable delamination is assigned, see Table 1. To verify the results, we compare to a reference model made with standard LS-DYNA elements and materials (corresponding to those in the user element).

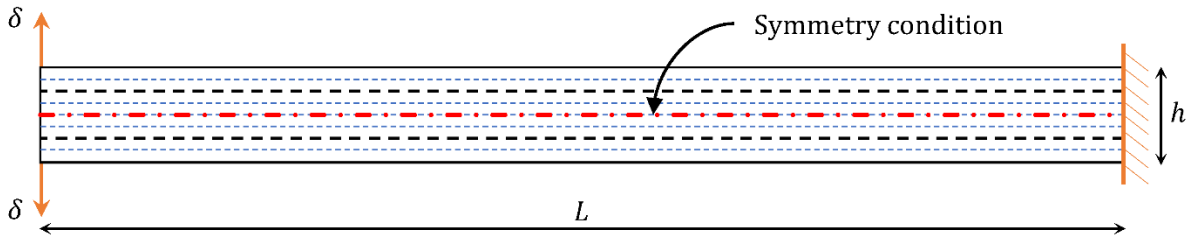

Figure 3: Illustration of TCB example. The beam has a length of L = 50 mm and thickness of h = 2.048 mm.

Parameter	Value	Parameter	Value
E_1	100 GPa	σ_I	20 MPa
E_1, E_1	10 GPa	$\mathcal{G}_{Ic}$	400 J/m^2
G_{12}, G_{13}	5 GPa	k_I	5 GPa
G_{23}	4 GPa	σ_{II}	4 GPa
ν_{21}, ν_{31}	0.025	$\mathcal{G}_{IIc}$	0.025
ν_{32}	0.25	k_{II}	0.25
ρ	1600 kg/m^3	ρ_{CZ}	1600 kg/m^3

Table 1: Ply and interface material properties used in the TCB example.

In Figure 6 we have plotted the reaction force versus displacement for the reference, a non-adaptive and an adaptive simulation. In the non-adaptive simulation, the user element is refined from the start. In the adaptive simulation the refinements are made at a displacement level of δ = 0.0023 mm, corresponding to a CZ failure initiation level of 1.0 in the reference. No stabilisation was needed for the adaptive case and in Figure 5 the deformed beams are shown.

If we postpone the refinement to δ = 0.005 mm, i.e. long after the reference CZ element failure initiation, there will be non-physical oscillations, as shown in Figure 7. By introducing the strong refinement over 0.01 ms, the solution is stabilised. We want to mention that the start positions and velocities of the extra nodes during refinement is likely rather good due to the trough-the-thickness homogeneity. Instead, oscillations are the result of refining past the correct failure initiation level of the elements.

The average computational cycle time for the unrefined user element is 1.84 ms, while the (strongly) refined user element has a cycle time of 2.18 ms. This results in a computational save of 16% when the refinements are not active.

The main message from this example is that for many practical applications it might not be necessary to stabilise the refinement. If non-physical oscillations occur there is the possibility to stabilise the solution. Nevertheless, a better option is likely to make sure that the refinement is made before failure should have initiated.

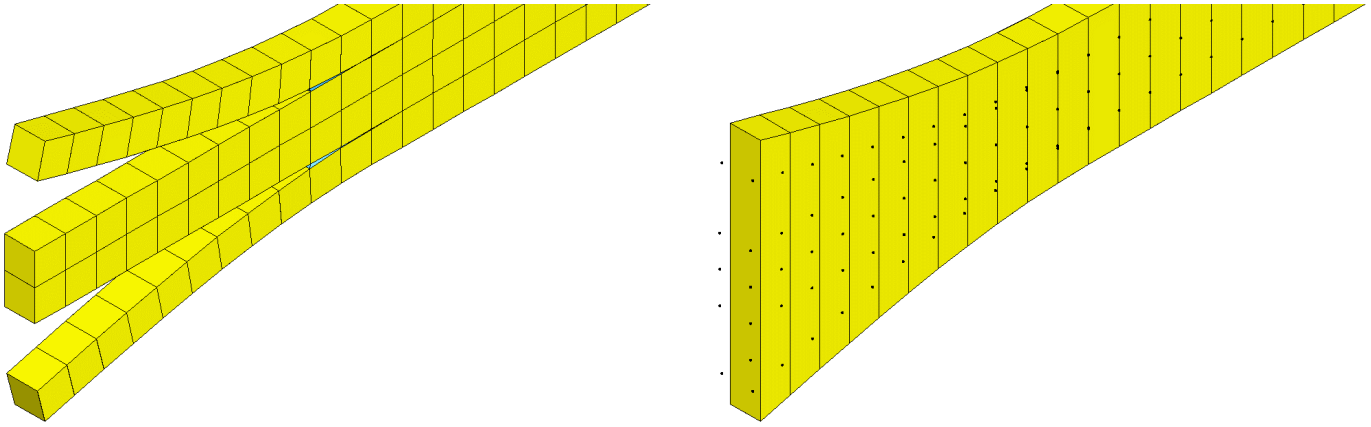

Figure 4: Deformed reference (left) and user element (right) beam at δ = 0.9 mm displacement. The nodes which define the internal subelements in the user elements are visible as black dots.

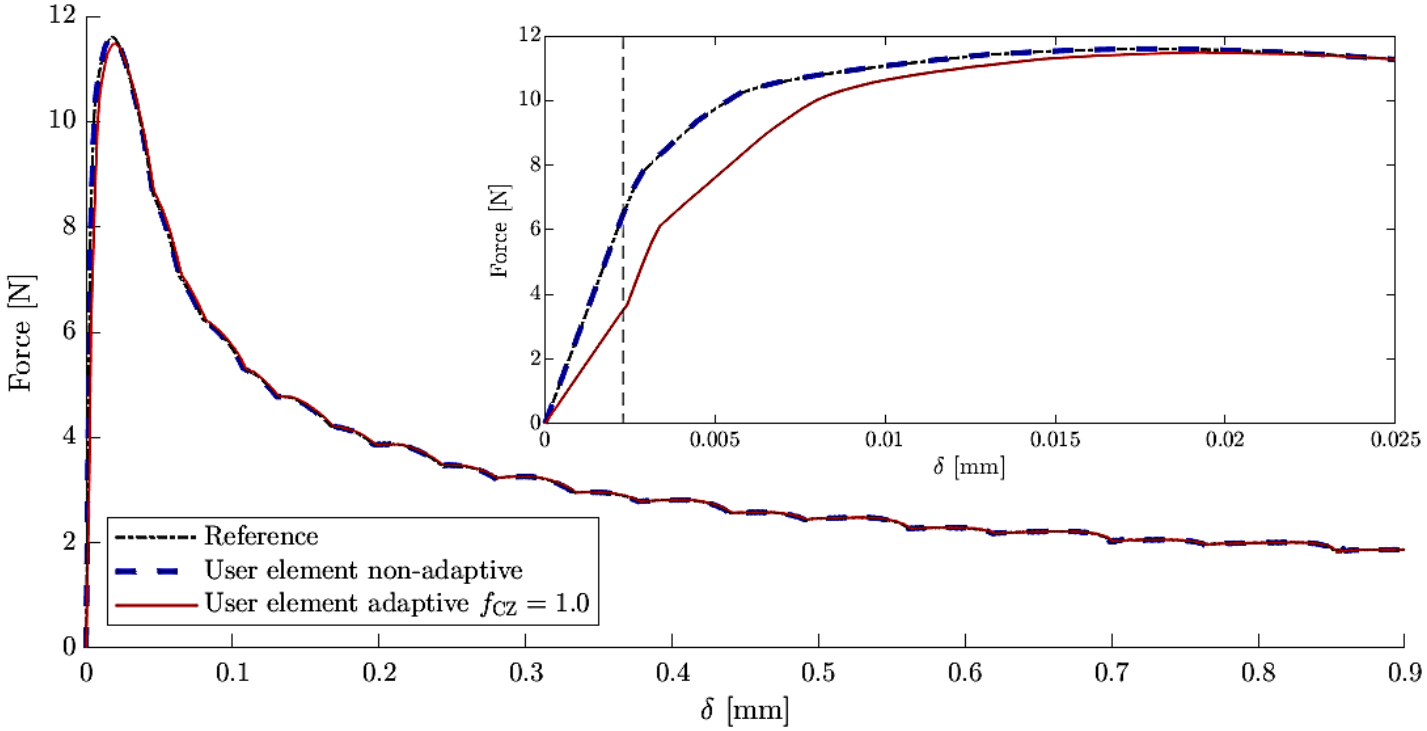

Figure 5: Load-displacement curves for TCB example. The user element with an initial refinement matches the reference exactly. The adaptive simulation shows a lower initial stiffness, but when refinements are activated at δ = 0.0023 mm (vertical line in zoom), the curve joins the other.

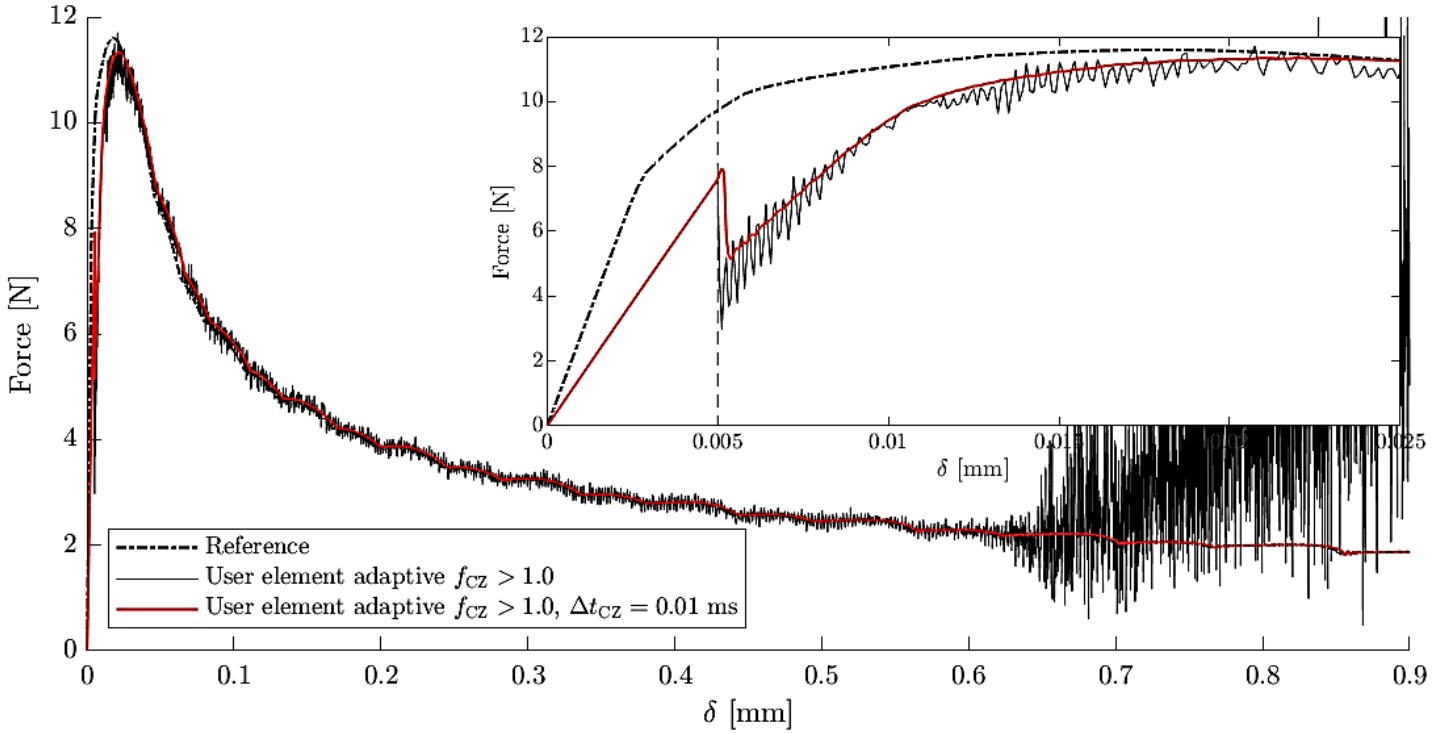

Figure 6: Load-displacement curves for TCB example. If the refinements are activated to late (vertical line in zoom), there will be an abrupt jump in the reaction force followed by non-physical oscillations. By introducing the strong refinement over 0.01 ms the solution is stabilised.

Matrix-crack induced delamination

In a second example we have simulated matrix-crack induced delamination in the four-point bending experiment performed by Mortell et al [8]. The material properties are taken from Reiner et al [9].

In Figure 8 we compare the load-displacement curves for simulations performed using a high-fidelity model with standard LS-DYNA elements and our adaptive user element with and without stabilisation. The user element is refined in the bottom 0/90-interface when the matrix failure index reaches unity in the lowermost layer. Damping is applied for 0.2 ms (corresponding to 0.08 mm displacement change), which is followed by the introduction of the strong refinement during 0.02 ms.

The computational save between the refined and unrefined user element is only 8% in this example.

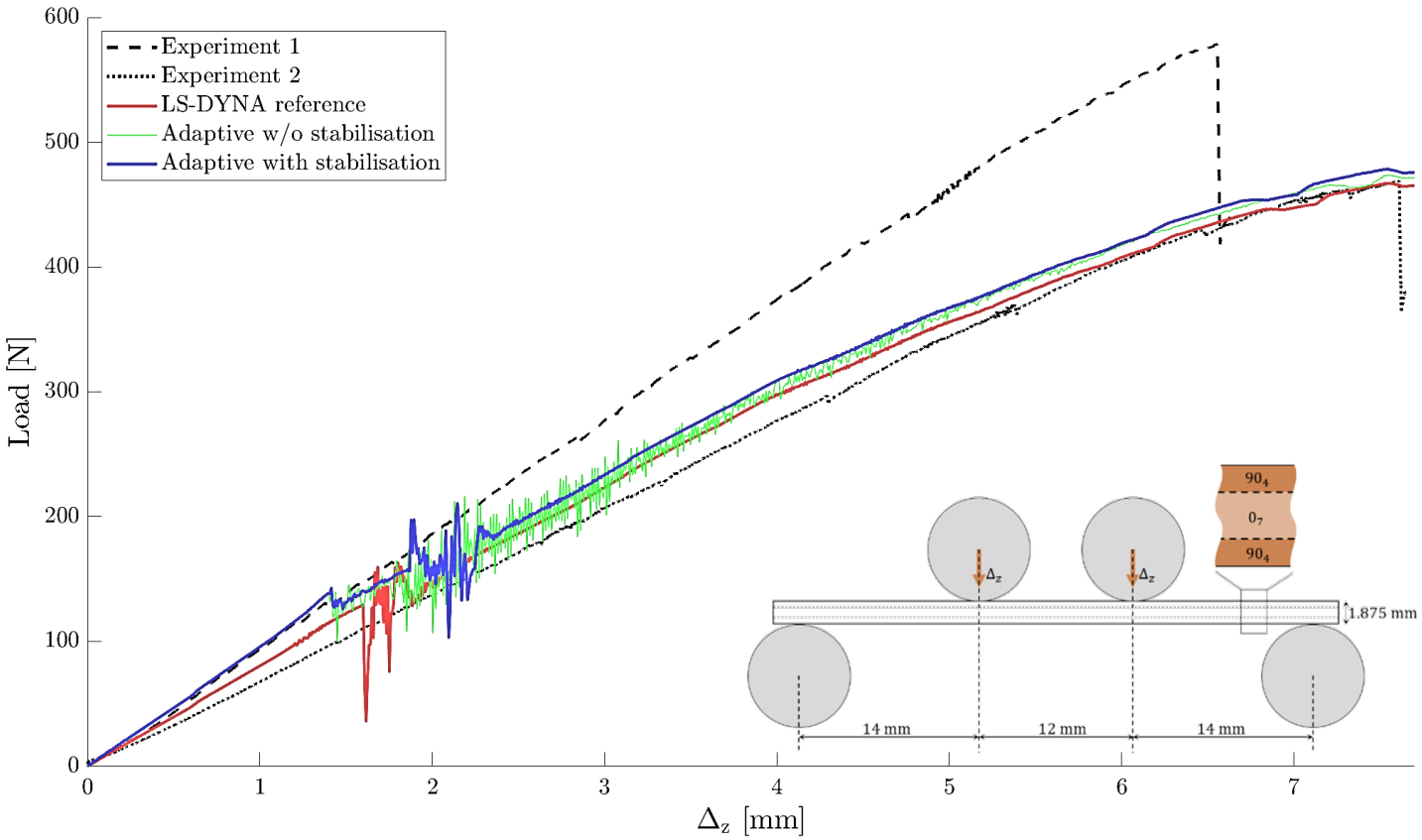

Figure 7: Load-displacement curves from simulation of a four-point bending experiment [4] where delaminations initiate from transverse matrix cracks. The adaptive simulations compare well with the reference high-fidelity model, however, without stabilisation unphysical oscillations will occur.

Conclusions

We have implemented an adaptive enrichment method for modelling of multiple and arbitrarily through-the-thickness refinements and delamination cracks using an ESL shell model in LS-DYNA. The method includes stabilisation with consideration to the instabilities that can occur when refining elements during a simulation in an explicit solver.

Using our adaptive method, we can reproduce similar results as a high-fidelity model while saving computational effort since all DOF are not present from the beginning of the simulation. While we currently only show a slight computational efficiency gain, we want to remind that only the costs of the element-internal computations are compared. Furthermore, computational efficiency is problem specific and if large parts of a model can consist of efficient ESL element while only critical sections are refined to LW elements, higher computational efficiency will be achieved. Therefore, we believe that the use of adaptive refinements will enable computationally efficient large-scale predictive crash simulations of laminated FRP. This will in the long run help to develop crash structures made of laminated FRP.

References

[1] ERTRAC, "European Roadmap Safe Road Transport," 2011.
[2] J. Främby, M. Fagerström, and J. Brouzoulis, "Adaptive modelling of delamination initiation and propagation using an equivalent single-layer shell approach," *Int. J. Numer. Methods Eng.*, vol. 112, no. 8, pp. 882–908, 2017.
[3] J. Främby, M. Fagerström, and J. Karlsson, "An adaptive shell element for explicit dynamic analysis of failure in laminated composites – Part 1: Adaptive kinematics and numerical implementation," *Submitt. to Eng. Fract. Mech.*, 2020.
[4] J. Främby and M. Fagerström, "An adaptive shell element for explicit dynamic analysis of failure in laminated composites – Part 2: Progressive failure and model verification and validation," *Submitt. to Eng. Fract. Mech.*, 2020.
[5] S. Costa, "Physically based constitutive models for crash of composites," Chalmers University of Technology, 2019.
[6] W. Liu, Q. D. Yang, S. Mohammadizadeh, and X. Y. Su, "An efficient augmented finite element method for arbitrary cracking and crack interaction in solids," *Int. J. Numer. Methods Eng.*, vol. 99, no. 6, pp. 438–468, 2014.
[7] T. Menouillard and T. Belytschko, "Smoothed nodal forces for improved dynamic crack propagation modeling in XFEM," *Int. J. Numer. Methods Eng.*, vol. 84, no. 1, pp. 47–72, 2010.