Investigation into the Application of Smart Aggregates in Monitoring Composite Precast Concrete Beams Made Continuous

Vincent Ponson
4270827

Investigation into the Application of Smart Aggregates in Monitoring Composite Precast Concrete Beams Made Continuous

By

Vincent Nicolas Ponson

In partial fulfilment of the requirements for the degree of

Master of Science
In Civil Engineering, Structural Engineering

At the Delft University of Technology

Supervisor:	Dr. ir. Y. Yang,	TU Delft
Thesis committee:	Dr. F. Kavoura	TU Delft

Acknowledgements

Foremost, I would like to express my sincere gratitude to the members of my thesis committee at TU Delft, Dr. ir. Yuguang Yang and Dr. Florentia Kavoura, for their continuous guidance, constructive feedback, and valuable insights throughout the course of this research

I would like to extend my appreciation to Dr. Yuguang Yang for proposing this challenging and engaging thesis topic and for his continuous support, particularly in the final stages of this work, helping me bring this thesis to completion.

I am grateful to Mohammed Ibrahim for his support in familiarising me with the experimental beam tests and the underlying structural behaviour, as well as for his assistance during the thesis writing process.

I would also like to thank Hao Cheng for his guidance in understanding Smart Aggregate technology and ultrasonic techniques, and for his support with the MATLAB implementation and data processing.

I would also like to extend my appreciation to everyone at Stevin Lab 2 who supported me during this project, even if not mentioned here by name. All of these contributions have been invaluable to the successful completion of this MSc thesis, for which I am deeply grateful.

Finally, I would like to thank my parents for their unwavering support, encouragement, and belief in me throughout my studies, which made it possible for me to reach this milestone.

Abstract

Ensuring the long-term reliability of ageing concrete infrastructure has increased the demand for effective Structural Health Monitoring methods capable of detecting internal damage at early stages. One promising technique involves the use of Smart Aggregates (SA) to monitor changes within the structure non-destructively. This study focuses on their application in composite precast concrete beams made continuous, particularly at the concrete-to-concrete interface between the precast inverted T-beam and the cast-in-situ topping layer. This interface is vulnerable to degradation due to shear forces and stress concentrations, which can result in internal defects such as microcracking or interface separation—forms of damage that are important to monitor, yet difficult to assess in practical in-situ conditions.

The aim of this study is to evaluate how SA-based ultrasonic monitoring can be effectively utilised to detect, localise, and characterise interface delamination in composite precast concrete beams made continuous. Focus is placed on assessing the complementary capabilities of Ultrasonic Pulse Velocity (UPV), in terms of spatial localisation, and Coda Wave Interferometry (CWI)-derived indicators, in terms of sensitivity to early-stage damage. Additionally, the study evaluates how ray-path and tomographic visualisation approaches support the interpretation of these ultrasonic techniques.

UPV is applied by analysing changes in wave arrival time to determine variations in propagation velocity, enabling detection of discontinuities along defined transmission paths. In contrast, CWI utilises the sensitivity of microcracking and stress redistribution to detect incremental changes within the material. From these methods, three ultrasonic indicators are derived: relative velocity from UPV, and the correlation coefficient (CC) and relative velocity change (ε) from CWI. These indicators are analysed through ray-path representations, which preserve path-based physical meaning. Additionally, UPV results are reconstructed into a tomographic visualisation to provide a spatial representation of internal structural changes, from which the interface time interference indicator is derived to assess interface delamination.

The experimental program consists of large-scale precast girders made continuous, embedded with SA and subjected to staged loading. The experimental work was conducted as part of a broader research campaign at TU Delft, in which the physical experiments were carried out by M. Ibrahim. Ultrasonic measurements are collected and processed using the described methods, and the resulting indicators are compared with Digital Image Correlation measurements for validation. This approach enables assessment of the capability of ultrasonic indicators to detect, localise, and quantify damage progression at the interface.

The results show that the ultrasonic indicators provide distinct yet complementary insight into structural behaviour. CWI-based indicators demonstrate high sensitivity to early-stage changes: ε responds to early disturbances prior to visible damage, while CC provides a clearer and more consistent indication of crack initiation. In contrast, UPV-derived relative velocity correlates strongly with developed cracking and provides reliable localisation along transmission paths, particularly for sensor pairs oriented to capture flexural and shear cracks. However, sensor pairs crossing the interface show reduced capability in distinguishing specific crack types, indicating limitations in isolating interface delamination independently.

Ray-path results show strong agreement with DIC observations in terms of crack initiation, localisation, enabling direct interpretation along propagation paths. Tomographic results provide a spatial overview of damage distribution and indicate potential for identifying interface disturbances through the dt indicator at early load stages; however, reconstruction limitations and numerical sensitivities reduce reliability in consistently representing damage magnitude and progression.

Contents

Chapter 1 – Introduction

1.1 – Background

As civil infrastructure across the Netherlands—and globally—continues to age, the importance of ensuring its long-term functionality and safety has become a central concern in engineering practice (Alampalli & Ettouney, 2008). Bridges, in particular, form critical nodes in transportation networks, and any compromise in their structural integrity can lead to significant economic, logistical, and safety repercussions (Farrar & Worden, 2007). Among the various bridge types, composite precast girder bridges made continuous—especially those utilizing inverted T-beam girders combined with cast-in-situ concrete decks—have become a staple of Dutch bridge construction since the 1970s (Borges et al., 2024; Calavera et al., 2004).

Historically, girder bridges were designed as simply supported systems, with load-bearing supports located at both ends of each girder (Mattock & Hawkins, 1972). This approach offered simplicity in design and construction but imposed limitations on span length and introduced challenges in achieving material efficiency (McHenry & Mattock, 1960). As technology advanced, a shift toward continuous girder systems emerged. In this configuration, the bridge deck is supported at multiple points along its span, which distributes moments more evenly resulting in longer, more slender structures with fewer bearings and joints. While offering structural efficiency and redundancy, continuity also introduces complexities in moment redistribution and increased negative moments at interior supports. This evolution underscores the need for careful interface design where precast girders and cast-in-place decks meet.

Understanding and accurately assessing interface behaviour is essential, not only for evaluating the overall strength of the bridge but also for maintaining and extending its service life. Studies by P. M. D. Santos et al. (2014) and Davaadorj et al. (2020) highlight how interface conditions—such as surface roughness and bonding quality—directly affect structural performance. Poor shear transfer can lead to slippage and reduced load capacity, particularly under repeated loading. Thus, monitoring this interface is vital for detecting early signs of deterioration and ensuring long-term bridge safety and performance. Traditional assessment methods—such as visual inspections and surface-level measurement techniques like Digital Image Correlation (DIC)—offer useful information about external deformation and cracking (Agdas et al., 2015). However, these methods are inherently limited in detecting internal changes, particularly at embedded regions such as the girder-deck interface, and are often impractical for long-term, in-situ monitoring. This limitation necessitates a shift toward integrated systems capable of capturing internal structural changes in real-world settings

In recent years, Structural Health Monitoring (SHM) has emerged to address these limitations. SHM uses integrated sensor systems to continuously or periodically monitor structural condition throughout service life. Unlike traditional inspections, SHM provides real-time performance data, enabling early detection of subtle damage. As Farrar & Worden (2007) note, SHM is not merely about sensing damage—it also involves processing complex data sets to assess structural health, reliability, and future performance. By enabling predictive, condition-based management, SHM allows more efficient resource use and helps prevent catastrophic failure. Moreover, Alampalli & Ettouney (2008) underscore SHM's role in bridge security, stating that a growing number of bridge owners are deploying SHM systems to enhance resilience. Their research highlights how continuous monitoring provides actionable insights, by doing so enabling data-driven decision-making.

Among SHM innovations that provide insight into the structural integrity of continuous girder bridges—particularly at the interface between the girder and deck—Smart Aggregates stand out (Cheng et al., 2023). These piezoelectric sensors, embedded within the concrete, can both transmit and receive ultrasonic waves—compressional pulses that travel through the material and reflect its internal condition (Karaiskos et al., 2015). By analysing ultrasonic wave propagation, changes in stiffness can be inferred, cracks can be detected, and damage progression can be monitored. With dimensions and mechanical properties similar to those of coarse aggregates, SA integrate seamlessly into the concrete matrix with minimal disturbance (Cheng et al., 2023).

Smart Aggregates on their own function as embedded transducers that generate and receive mechanical waves within a medium, while the interpretation of the recorded waveforms is performed through dedicated signal processing techniques. Among these, Coda Wave Interferometry (CWI) is particularly valuable due to its sensitivity to microstructural changes within the material (Fröjd & Ulriksen, 2017; Snieder, 2006). It can detect subtle internal alterations, such as early-stage cracking, long before such damage becomes visible on the surface. Complementing this, Ultrasonic Pulse Velocity (UPV) assesses the velocity at which ultrasonic waves travel through the concrete. Variations in wave velocity are directly linked to changes in material integrity and can indicate the presence of cracks or internal deterioration (Karaiskos et al., 2015). Together, these methods provide a powerful framework for analysing ultrasonic signals and assessing the health of concrete structures at subsurface level than traditional techniques allow. To make the results more intuitive for readers, visualization techniques like tomographic imaging and ray-path reconstruction are required (Camassa et al., 2020). Tomography reconstructs internal property distributions based on wave data, creating visual damage maps. Ray-path imaging, using time-of-flight and amplitude data, complements tomography in highlighting internal anomalies. Together, these tools convey complex internal conditions in intuitive formats for engineers and stakeholders.

The implementation of SA and ultrasonic analysis represents a shift toward proactive infrastructure management. Rather than responding to visible damage detected during periodic inspections, this approach enables possible early detection, and a potential new method to examine the state of the girder-deck interfaces. This thesis focuses on SA based monitoring of such interfaces, using CWI and UPV, alongside tomographic and ray-path imaging-techniques. By comparing internal damage maps with external DIC measurements, this research hopes to add value to embedded SHM systems.

1.2 – Problem Statement and Research Aim

Despite progress in surface-level techniques, there remains a lack of practical and reliable methods for in-situ monitoring of internal behaviour—especially at critical interfaces in precast concrete structures. The interface plays a fundamental mechanical role in ensuring composite action, enabling shear transfer and continuity of force flow across the structure; degradation of this interface directly compromises load transfer mechanisms. The inability to detect early-stage damage, such as microcracking or debonding, limits the effectiveness of maintenance strategies and increases the risk of sudden failure. Additionally, SA have shown promise in capturing internal wave responses, their implementation remains limited both in laboratory experiments and in real-world structural monitoring. Moreover, the post-processing of ultrasonic data into image-based representations is still evolving and presents opportunities for further refinement and enhanced interpretability.

This thesis aims to contribute to this growing body of knowledge by focusing on the application of SA in monitoring the interface behaviour of girder bridges made continuous. By analysing ultrasonic measurements using CWI and UPV techniques and employing strategies to interpret

internal structural changes through tomographic and correlation-based imaging, this research seeks to identify early indicators of crack development and potential interface failure. Emphasis is placed on assessing the location, spatial extent, and relative magnitude of these indicators, in order to evaluate the ability of ultrasonic measurements to not only detect damage but also provide meaningful information on its development and localisation within the structure.

1.3 – Research Questions and Sub-questions

How can Smart Aggregate technology be effectively utilized to monitor cracking and interface delamination behaviour at critical interface regions in composite precast girder bridges made continuous?

Sub-Questions:

1. What information can be obtained from Smart Aggregate–based ultrasonic measurements regarding crack and interface behaviour in composite precast girder bridges made continuous, and how do the resulting ultrasonic indicators provide distinct yet complementary insight on actual crack and interface behaviour of the structure?
2. How can Smart Aggregate–based ultrasonic measurements be processed and visualized in a clear and meaningful manner to highlight the relevant ultrasonic indicators and support their interpretation with respect to crack and interface behaviour?
3. To what extent can ultrasonic indicators which are derived from Smart Aggregate measurements be validated against established measurement techniques to substantiate interpretations of cracking and interface delamination?

1.4 – Approach

The experimental campaign features a laboratory-scale beam test designed to replicate the shear-critical interface conditions typical of composite precast girder bridges, as highlighted by Ibrahim et al. (2022), who underscore the challenges posed by such configurations. This test is part of a broader series of experiments conducted at TU Delft that collectively aim to investigate structural behaviour and performance in composite precast bridge made continuous. The physical experiments were originally carried out by M. Ibrahim, and to which this thesis contributes through detailed analysis and interpretation of the interface region of the precast concrete girder. The test specimen consists of a composite concrete beam made up of a precast inverted T-beam girder and a cast-in-situ topping layer, simulating a monolithic connection. SA are embedded at strategic locations adjacent to the intermediate support, where shear stresses and deformation concentrations are expected to be highest. This placement is intended to improve the likelihood of capturing interface splitting, crack initiation, and other critical damage developments.

The ultrasonic data collected from the SA are post-processed to produce image-based visualizations that reflect internal changes. These images are generated using a modified implementation of a signal-mapping algorithm originally developed by H. Cheng at TU Delft, adapted to account for the geometry and sensor configuration specific to this study (Cheng et al., 2023). While the MATLAB code was originally implemented by F. Zhang, the underlying mapping algorithm was developed by H. Cheng and further modified for this study (Appendix III – MATLAB code). In addition to this adapted algorithms, supplementary mapping algorithms will be developed to support velocity-based tomographic image generation. The goal of this processing step is to translate variations in wave behaviour into images capable of identifying localized changes in waveform, such as those caused by microcracks, delamination, or and flexural and shear cracking. While image-based monitoring offers a powerful representation of internal damage, the method remains under active development and provides substantial room for refinement in terms of spatial resolution, reliability, and interpretability.

To assess the reliability and effectiveness of the Smart Aggregate–based monitoring approach, the ultrasonic indicators are evaluated through comparison with independently observed crack development and deformation patterns obtained from Digital Image Correlation (DIC), which were provided by the tests and analyses preformed by M. Ibrahim. This comparison provides an experimental reference against which the sensitivity, interpretability, and limitations of the ultrasonic measurements can be assessed. The combined application of CWI and UPV enables evaluation of both early detection capability and localisation performance within the monitored region through the calculation of their respective ultrasonic indicators. Furthermore, the resulting indicators are presented through ray-path and tomographic representations. In this context, ray-path representations preserve the direct relationship between wave propagation and structural response along individual sensor paths, while tomographic reconstruction provides a spatially continuous representation of internal structural changes by combining multiple transmission paths. These visualisation approaches are used to examine how different representation strategies influence the interpretation of structural behaviour and to identify the most informative means of presenting ultrasonic indicators.

1.5 – Structure of this thesis

The thesis report is divided into several chapters discuss the following aspects of the report:

- Chapter 2 provides the theoretical background SA and its role and techniques used in SHM, establishing the framework for interpreting interface behaviour. This chapter addresses the first research sub-question.
- Chapter 3 presents the experimental methodology and explains the techniques used to monitor the composite precast girder beam made continuous using SA, this chapter addresses the second research sub-question.
- Chapter 4 evaluates each ultrasonic indicators through direct comparison with structural behaviour observed using DIC, this chapter contributes to the third research sub-question.
- Chapter 5 further analyses the ultrasonic indicators through their representation in ray-path and tomographic visualisations; this chapter also contributes to the third research sub-question.
- Chapter 6 presents the final conclusions of the study and provides answers to the research questions.

Chapter 2 – Literature Review on Ultrasonic based Monitoring of Concrete Interfaces

This chapter aims to provide an in-depth overview of the existing body of knowledge and research in the field of structural health monitoring (SHM) in concrete structures, particularly focusing on technologies and techniques relevant to monitoring the integrity of concrete girder bridges, which employ Girders made continuous. The goal is to familiarize the reader with foundational concepts, advancements in monitoring techniques, and the latest applications of Smart Aggregates (SA) and ultrasonic-based methods.

The chapter is structured into three parts. The first part introduces Composite Precast Girder made continuous, detailing their design and structural behaviour, including the mechanism of interface behaviour and previous research on these topics. Then transitions into SHM techniques, discussing digital imaging correlation, ultrasonic-based monitoring, and the challenge of bridging laboratory methods with real-world, in-situ monitoring. The second half of the chapter delves into Smart Aggregate-based monitoring, explaining the principles behind these systems, their role in SHM, and the data processing techniques used, such as Ultrasonic Pulse Velocity (UPV) and Coda Wave Interferometry (CWI). Additionally, this section discusses enhanced image-based monitoring techniques like tomography and their application in SHM. Finally, the chapter concludes with a summation on the possible application of these techniques and outlines the objectives of the present thesis.

Through this review, this chapter highlights key gaps in the understanding of how advanced monitoring technologies can be optimized for the assessment of concrete structures, particularly in improving crack detection and structural integrity analysis.

2.1 - Interface Monitoring of Composite Precast Girder made Continuous

Interface monitoring of Composite Precast Girder made continuous involves assessing the condition of the joint between the precast beam and the cast-in-place deck. While a more detailed explanation is provided later, this technique is crucial for understanding structural performance, especially in aging infrastructure. In the Netherlands, these girders are widespread, and as structures continue to age, the importance of monitoring their interfaces will grow, supporting informed decision-making for maintenance and safety. Therefore, the development and implementation of new monitoring techniques will become increasingly vital.

2.1.1 – Composite Precast Girder made continuous

Continuous precast girder bridges come in many unique designs and applications, a commonly found type in the Netherlands is the Precast Girder made continuous, as seen in Figure 1. This girder type has played a crucial role in Dutch infrastructure, especially in highway bridges and viaducts, and has been widely used from the 1970s through to the present day (Calavera et al., 2004; Spanbeton BV, 2014). Due to its widespread application and aging condition, the precast girder made continuous has recently become the focus of increased scrutiny, including laboratory testing and in-situ monitoring efforts (M. S. Ibrahim et al., 2022; Vergoossen et al., 2022).

Figure 1, concrete bridge using girder beam, under construction (Haitsma Beton B.V., n.d.)

The beam system comprises several key components, namely the support system, beam girder and importantly for this thesis the interface. An example of a bridge system comprised of Precast Girder made continuous can be seen in Figure 2 and Figure 3, with the components highlighted.

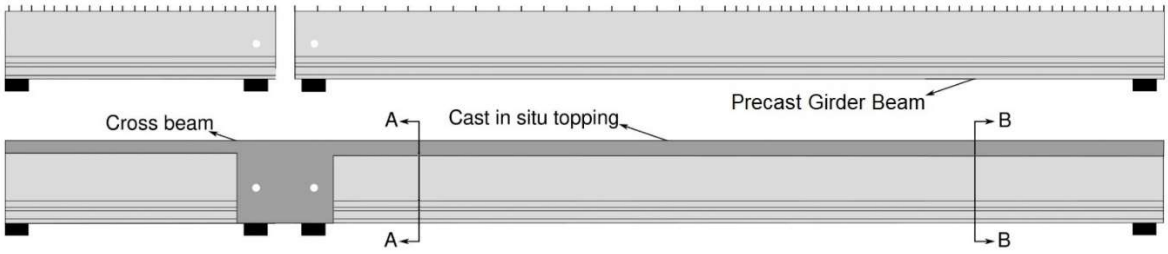


Figure 2, beam system with its components, adapted from (M. S. Ibrahim et al., 2025)

The support system typically includes piers and abutments, which transfer loads from the bridge to the foundation. In the past, this type of girder bridges typically utilized simply supported spans, where each beam was supported at its ends without continuity between spans (Kaar et al., 1960; Mirza & Furlong, 1985). In contrast, modern girder bridges now frequently employ a continuous support method, where the beams are connected across multiple spans to create a continuous structure. This change offers significant benefits in terms of structural behaviour and moment distribution.

By providing continuous support, the negative moments are introduced over the supports, effectively reducing the positive moments at mid-span. This results in a more balanced moment distribution, reducing peak moments at mid-span and allowing for more efficient use of materials. Consequently, the bridge structure can be lighter and more economical while maintaining or even enhancing its load-carrying capacity and durability (Burns, 1966; Calavera et al., 2004; Taha & Jia, 2018).

The beam girder can be further divided into several key components, most notably the precast T-beam and the in-situ cast concrete topping layer. In Figure 3 , the main elements of the girder are illustrated, including the T-shaped precast beam, internal reinforcement, prestressing tendons, the interface layer between precast and cast-in-place components, and the monolithic topping layer.

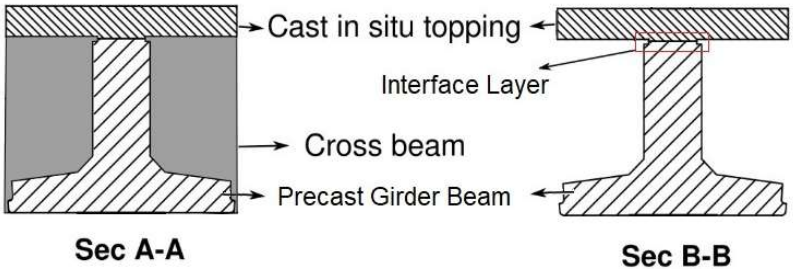


Figure 3, cross section A-A and B-B with components, adapted from (M. S. Ibrahim et al., 2025)

The T-beam is typically prestressed using tendons positioned near the bottom of the beam to counteract tensile stresses, thereby enhancing both structural strength and achievable span length. As a precast component, the T-beam is manufactured offsite under controlled conditions and then transported to the construction site. The topping layer, by contrast, is poured in-situ and serves to connect the individual girders into a continuous structural system. This multi step construction of the girder beam can be seen in the Figure 4 here below. This composite action enables effective load distribution across all beams, increasing stiffness and reducing differential deflection between adjacent girders.

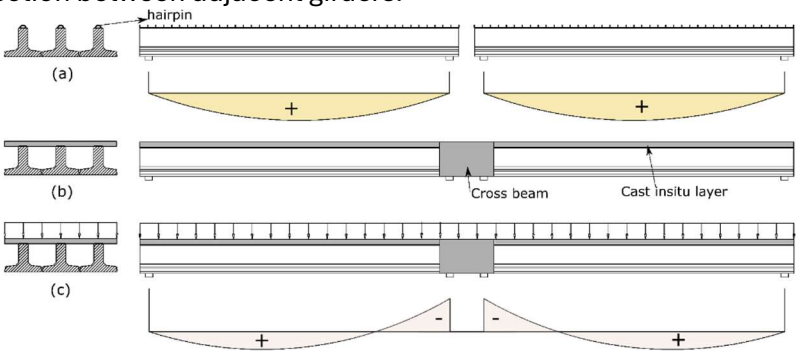


Figure 4, overview of the construction phases of the girder beam made continuous, with the change in moments distribution; (a) Precast inverted T girder before continuity (b) Continuous precast girder (c) response to traffic loading (UDL), adapted from (M. S. Ibrahim et al., 2022)

The region where the two main girder elements meet is called the interface, the interaction between the precast girder and in-situ topping layer is critical for the composite action of the bridge (Kaar et al., 1960). This region is often reinforced using shear connectors, to ensure effective load transfer and prevent separations or slips between the components, an example can be seen in Figure 5. Important design elements within this interface include reinforcing bars and hairpins that extend from the T-beams into the deck to provide additional tensile strength and ensure a robust connection.

Figure 5, interface and hairpins, taken from (M. S. Ibrahim et al., 2025)

The interface in a girder is a complex component, as it involves the interaction between varied materials and the transfer of forces between the T-beam and topping layer. This complexity has led to significant efforts by researchers to model and understand its behaviour accurately. Over the years, numerous models have been proposed to predict the interface's performance, particularly in terms of shear transfer, bond strength, and overall structural integrity (Davaadorj et al., 2020; Randl, 2013; P. M. D. Santos & Júlio, 2012; Walraven & Reinhardt, 1981). These models aim to capture the intricate mechanics of the interface, including factors such as slip, cracking, and load distribution.

Types of Concrete Interfaces

In concrete structures, interfaces can occur between various materials, such as steel-to-concrete, concrete-to-masonry, or concrete-to-concrete (Davaadorj et al., 2020). Among these, the concrete-to-concrete interface is particularly significant, especially in precast girder bridges. A concrete-to-concrete interface is the boundary between two concrete elements; this typically occurs when the two elements are cast at different times. The concrete-to-concrete interface

can be subjected to shear, tension, or compression forces, and its behaviour is often modelled to predict the performance of composite structures under various loading conditions.

Mechanisms of the Interface Behaviour

The bond strength at crucial aspect in a concrete-to-concrete connection, as it determines the effectiveness of load transfer between the two concrete elements (P. M. D. Santos et al., 2014; Walraven & Reinhardt, 1981). Factors like surface roughness, cleanliness, moisture content, and curing conditions significantly impact the quality of this bond. An important mechanism of at an interface is the shear transfer mechanism, in concrete interfaces this refers to the process by which shear forces are transmitted across the interface between two connected elements. This mechanism is critical in ensuring that the composite structure behaves as a single unit under load.

The concept of shear transfer originates from the need to maintain the structural integrity of elements like precast beams and slabs, where different concrete sections must work together. Shear transfer is facilitated through various components and mechanisms (Birkeland & Birkeland, 1966; Randl, 2013), including aggregate interlock, where the roughness of the interface allows for the interlocking of aggregates, friction between the surfaces, and dowel action provided by reinforcement crossing the interface. An often-used model is the saw-tooth model, which exemplifies the basic principles of shear transfer theory, as seen in Figure 6a (P. Santos & Júlio, 2010). This model accounts for the combined influence of reinforcement and external forces acting normal to the shear plane, merging them into a single resultant clamping force. Similarly, Figure 6b illustrates the transfer mechanisms as presented by Zilch and Reinecke (2000), highlighting how adhesion, friction, and reinforcement each contribute to shear resistance. The figure also emphasizes that these mechanisms are activated at different stages of slip or displacement—adhesion being mobilized first, followed by friction, and finally dowel action from the reinforcement as deformation increases. Understanding these mechanisms is vital for the design and analysis of composite and precast concrete structures, ensuring their safety and durability under operational loads.

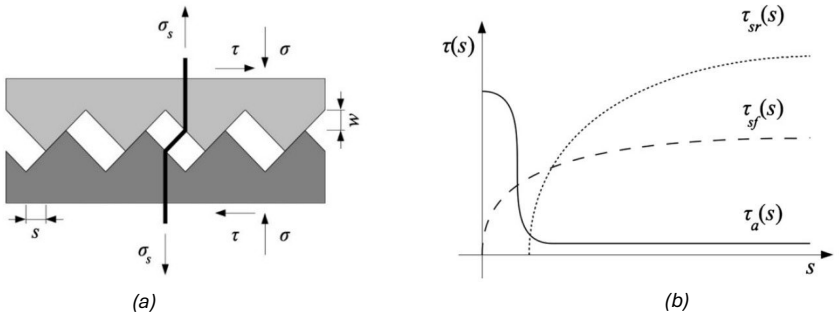


Figure 6, saw-tooth model (a), Load transfer mechanisms as present by Zilch and Reinicke (2000) (b), both taken from (P. M. D. Santos & Júlio, 2012)

Adhesion refers to the bonding between different materials at the interface (Davaadorj et al., 2020; P. M. D. Santos et al., 2014; Walraven & Reinhardt, 1981). The strength of adhesion is a critical factor in determining the load-carrying capacity and durability of the interface. Factors influencing adhesion include surface roughness, cleanliness, and the type of adhesive used. Furthermore, the adhesion only contributes to the mechanism as longs as the bond still holds and stops ones the bond is broken. Adhesive bonds can be used to join materials like concrete and steel or to attach reinforcement elements to beams.

Friction at the interface plays a significant role in load transfer and stability (P. M. D. Santos et al., 2014; Walraven & Reinhardt, 1981). Frictional forces can help resist slippage and improve the effectiveness of mechanical connections. There are two main sources of friction, the presence of reinforcements or connectors and internal compressive stresses.

Reinforcement at the interface involves adding additional elements to enhance the load-carrying capacity and improve performance. Reinforcement provides an axial force clamping the two elements tighter, at the interface (Mattock & Hawkins, 1972). Besides the axial force and additional transfer mechanism occurs, namely 'Dowel action'. This refers to the shear transfer mechanism where reinforcing bars (dowels) across a concrete interface resist sliding between two connected elements. As the interface experiences shear, the bars bend and develop resistance, helping maintain the structural integrity of the composite or precast sections by preventing relative movement.

Each of these mechanisms contributes to the overall shear capacity of the interface, with their effectiveness depending on factors like the interface's roughness, the amount of reinforcement, and the compressive force normal to the interface.

2.1.2 – Notable Research and Experiments on Interface Behaviour

Over the years, researchers have conducted numerous experimental studies on Precast Girder made continuous, particularly focusing on the interface between precast beams and in-situ layers. These studies have significantly improved the understanding of shear transfer mechanisms and composite action. Leonhardt & Walther (1962) investigated shear transfer across roughened concrete-to-concrete interfaces, identifying the critical role of surface texture and reinforcement in enhancing shear capacity. Mattock & Hawkins (1972) studied shear-friction mechanisms in concrete interfaces, proposing design equations that account for reinforcement crossing the interface. Walraven & Reinhardt (1981) performed experiments at TU Delft on the aggregate interlock mechanism, emphasizing its contribution to interface shear resistance. Dall'Asta & Zona (2002) explored the influence of shear connectors on interface performance in composite beams, proposing new design guidelines for precast structures. P. Santos & Júlio (2010) examined the role of surface preparation and reinforcement layout on the shear strength of composite concrete sections, confirming the importance of interface detailing.

More recent studies have further emphasized the structural importance of the interface in precast girder bridges made continuous (M. Ibrahim & Yang, 2022). In these systems, the interface between the precast girder and the cast-in-place continuity concrete governs the transfer of shear forces and the development of composite structural behaviour. Experimental investigations have shown that deficiencies at this interface may lead to a loss of composite action, which can become a governing failure mechanism (Borges et al., 2024). Cracking may initiate due to tensile stresses generated by negative bending moments near the supports, while shear stresses can promote interface slip and progressive damage along the concrete-to-concrete interface (P. M. D. Santos et al., 2014). As this damage evolves, it may lead to debonding or delamination, reducing the effectiveness of shear transfer and compromising the structural behaviour of the system (Mohamad & Ibrahim, 2015). These findings highlight the importance of monitoring the interface region to detect early damage and assess structural performance.

2.1.3 – Structural Health Monitoring in Concrete Structures

Structural Health Monitoring (SHM) has become a vital component of modern civil engineering, particularly for ensuring the safety, functionality, and longevity of infrastructure systems (Farrar & Worden, 2007; Hoult et al., 2010). It refers to the continuous or periodic assessment of structures—such as bridges, buildings, and girders—through the use of embedded or surface-mounted sensors (Agdas et al., 2015; Alampalli & Ettouney, 2008). These systems are designed to detect changes in structural behaviour that could indicate damage or degradation. The implementation of SHM is especially critical concrete infrastructure, where failure could lead to serious safety risks and costly repairs.

At its core, SHM is a multi-stage process comprising sensing, data acquisition, signal processing, feature extraction, and decision-making. These steps are interlinked to form a feedback system capable of tracking structural performance over time and detect potential damage before it becomes critical. A key goal of SHM is to identify deviations from baseline or reference conditions(Agdas et al., 2015; Farrar & Worden, 2007; Guan et al., 2014). These deviations can manifest in several forms, such as shifts in resonance frequency, changes in strain distribution, wave attenuation, or increased signal noise—all of which may signal the onset of damage (Mardanshahi et al., 2025).

To achieve this, SHM systems employ a variety of techniques. For example, vibration-based monitoring can identify stiffness reductions due to cracking; acoustic emission methods can locate active crack growth; and ultrasonic testing can detect internal defects such as voids or delamination (Haque et al., 2022). Among these, ultrasonic-based techniques are particularly valuable in concrete structures due to their ability to penetrate dense materials and capture wave propagation characteristics.

Image-based Monitoring Techniques used for SHM

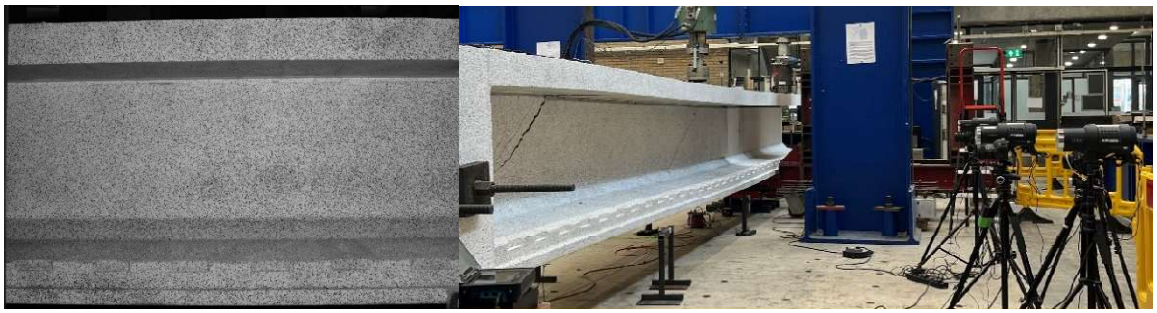
Visualization of a beam's internal and surface condition is instrumental aspect in SHM, as it enables the identification of damage mechanisms and structural anomalies that are not always detectable through traditional inspection methods. The ability to visualize defects—such as cracks, delamination, or debonding at interfaces—could play a vital role in assessing the beam's performance and safety. Effective visualization techniques not only help in localizing the damaged areas but also in quantifying the severity and progression of structural deterioration over time. Presenting SHM data in an accessible and easily interpretable format is crucial not only for researchers but also for structure owners and stakeholders (Farrar & Worden, 2007). Clear visualization and analysis can help bridge the gap between technical results and practical decision-making, making it easier for non-experts to understand structural conditions and justify maintenance actions. This clarity supports broader adoption of SHM systems by demonstrating tangible safety and economic benefits, which is essential for real-world implementation beyond academic research.

Some techniques associated with monitoring include Digital Image Correlation (DIC), which captures surface strain, and ultrasonic-based imaging, which reconstructs internal variations using wave propagation data. These visualization methods provide critical insights into structural behaviour, enabling real-time monitoring and predictive maintenance.
One such technique that utilizes ultrasonic-based methods is the SA, which integrate piezoelectric materials within the concrete matrix, enable internal wave generation and reception—making them highly suitable for monitoring stress changes, crack initiation, and propagation in situ (Cheng et al., 2023; Song et al., 2008). A more in-depth description of SA, including their working principles and applications in structural health monitoring, can be found in 2.2.1 – Working Principle of Smart Aggregates.

Existing Optical-Based Monitoring Techniques

An established monitoring technique— DIC—was employed in this thesis. DIC is a non-contact optical measurement technique widely used to analyse surface deformations (Pan et al., 2009; Schreier et al., 2009). It operates by comparing digital images of an object's surface captured at different states, such as before and after applying a load. In structural analysis, particularly for beams, DIC enables researchers to study deformation behaviour under applied forces. Its ability to detect localized phenomena, such as crack initiation and propagation, offers a comprehensive assessment of a structure's integrity and performance under load.
The DIC process begins with the preparation of the object's surface, typically applying a high-contrast speckle pattern to facilitate accurate tracking. Cameras capture images of the surface in its initial, undeformed state and during subsequent loading conditions. Advanced algorithms

divide these images into smaller subsets and track the movement of speckle patterns between the undeformed and deformed states using cross-correlation techniques. This tracking produces the *displacement field*, which represents how each subset of the surface has moved relative to its original position, an example of a *high-contrast speckle pattern* can be seen in Figure 7. By analysing these displacements, DIC provides a detailed understanding of the surface deformation under load. From this displacement data, the strain field is calculated through numerical differentiation, this procedure provides further insights into the material's deformation under load.



(a) *(b)*

Figure 7, example of a a high-contrast speckle pattern (a), test setup for a DIC experiment (b)

2.1.4 – Ultrasonic-based Monitoring

Ultrasonic-based monitoring has emerged as a reliable and versatile technique in SHM for concrete structures, particularly in detecting internal anomalies that may not be visible on the surface (Cheng et al., 2024; Singh et al., 2022). These methods utilize the propagation of mechanical waves—elastic waves—through a structure to identify both irreversible changes (i.e., damage-induced) and reversible changes (i.e., elastic) in the medium (Planès & Larose, 2013). Their ability to monitor stress distribution, micro-cracks, and voids makes them indispensable for structural integrity assessment (Song et al., 2008).

Elastic Waves in Concrete

Elastic waves are mechanical waves governed by the constitutive relation of the medium. The behaviour of these waves can be described mathematically by the elastic wave equation, which is derived from Newton's second law of motion and Hooke's law of elasticity, linking the displacement of particles in the medium to the stress and strain within the material.
When a medium is disturbed, it will try to restore itself to its equilibrium state and this generates a so-called restoring force. These forces result in oscillations that propagate through the medium as waves. If one then takes an infinitesimal segment apply Newton’s Second Law and Hooke’s law to describe the restoring force, one can derive the wave equation. The theory also accounts for wave reflection, refraction, and attenuation as elastic waves encounter different interfaces or anomalies within the material. This theoretical framework is essential for understanding how elastic waves interact with various internal structures.

Types of Waves

Elastic waves can be categorized into two types: body waves and surface waves. Body waves are a type of elastic waves that propagate through the interior of a medium, whereas surface waves propagate along the free surfaces or interface. In relation to SA, the sole focus is on the body waves, as SA are located and generate waves within a structure(Achenbach, 1999).

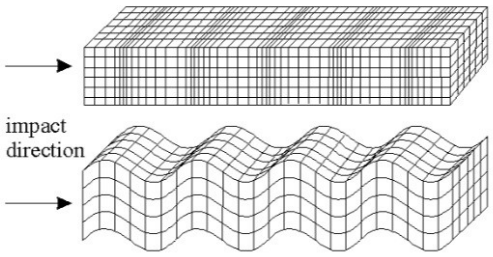


Figure 8, Schematic illustration of P-waves (top) and S-waves (bottom) propagating through a systematic matrix, adapted from (Hassani et al., 2001)

Body waves can be further categorised into two modes, the P-waves (longitudinal or compressional waves) and S-waves (transverse or shear waves). The main difference between these two modes is the direction in which the particles move, also known as the polarization direction, in relation to the wave propagation direction. For P-waves, the polarization direction is parallel to the wave propagation direction, whereas S-waves' polarization directions are perpendicular to the wave propagation direction. Furthermore, P-waves have a higher velocity and can travel through both liquid and solids. In contrast S-waves velocities are slower and these waves can only travel through solids.

2.1.5 – Bridging the Gap Between Laboratory and In-Situ Structural Health Monitoring

While laboratory-based studies have greatly advanced our understanding of concrete beam behaviour, applying these insights to in-situ monitoring remains challenging. Real-world conditions introduce variables such as environmental noise, fluctuating loads, and accessibility constraints, which are absent in controlled lab environments. These factors create uncertainties that complicate long-term structural assessments and limit the effectiveness of traditional monitoring techniques.

Over the years, researchers have conducted numerous experimental studies on SHM, focusing on long-term and/or in-situ monitoring, and implementation of new techniques and technologies. These studies have significantly improved the understanding of external and internal condition of a structure. Farrar & Worden (2007) outlined key definitions, methods, and challenges in structural health monitoring, emphasizing damage detection, sensor use, and data-driven analysis strategies. Hoult et al. (2010) presented the deployment of a wireless SHM system using strain gauges to monitor the Ferriby Road Bridge over an extended period. The research demonstrates the effectiveness of wireless sensor networks in capturing strain data. Haque et al. (2022) explored the use of accelerometers for indirect SHM by analysing vehicle-induced vibrations to detect and quantify damage in bridge structures. Aguero et al. (2023) presented an Augmented Reality system integrating Wireless Smart Sensors to visualize real-time structural displacements, improving field assessments and decision-making in SHM.

Recent advancements, such as the integration of Smart Aggregates (SA), offer a transformative solution for in-situ SHM (Cheng et al., 2023; Song et al., 2008; Yan et al., 2017). Embedded directly into concrete during casting, these sensors eliminate the need for surface mounting or external wiring. Their ability to both generates and receive ultrasonic waves enables active monitoring of internal stress distribution, crack formation, and material degradation in real-time—even under harsh environmental conditions. By supporting techniques like Ultrasonic Pulse Velocity (UPV) and Coda Wave Interferometry (CWI), SA enhance the precision of internal damage detection. However, despite their promise, most applications have been limited to controlled environments, highlighting the need for robust packaging, durable integration methods, and signal calibration strategies for effective deployment in real-world infrastructure.

2.2 – Smart Aggregates-based Concrete Monitoring

SHM has become an essential field in civil engineering, particularly for assessing the integrity and long-term performance of concrete structures such as beams (Farrar & Worden, 2007). Advances in sensing technology, particularly in non-destructive evaluation (NDE) methods, have transformed SHM by enabling real-time damage detection and predictive maintenance. Among these advances, SA have emerged as a possible tool for in-situ ultrasonic monitoring, enhancing the capability of SHM techniques (Song et al., 2008).
This chapter provides background information on the principles and applications of SA, as well as two ultrasonic-based data processing techniques in addition to a discussion on tomography for visualizing internal structural changes (Camassa et al., 2020). It also highlights the importance of the in-situ assessment method to beam health.

2.2.1 – Working Principle of Smart Aggregates

SA are embedded ultrasonic sensors for concrete infrastructure monitoring, to ensure the safety and durability of these structures, an example of a SA can be seen in Figure 9 here below. These SA have been increasingly utilised in experiments to enhance our understanding of the internal state and behaviour of structures, providing critical insights into their integrity and performance under various conditions (Song et al., 2008). They operate by emitting and receiving elastic waves, utilizing ultrasonic-based technology to assess internal changes. The dimensions of a SA are comparable to those of coarse aggregates; as a result, this sensor does not create a significant disturbance to the concrete matrix, preventing the creation of weak points or areas of high stress concentration. This allows SA to monitor a structure without compromising the integrity of the concrete, making them a desirable sensor for long-term monitoring of concrete structures (Singh et al., 2022).

(a) (b)

Figure 9, example of a SA sensor (a), taken from (Cheng et al., 2023), and a SA sensor in a beam before casting (b), adapted from (S. Zhang et al., 2022)

Principle of Smart Aggregates

SA operate based on the piezoelectric effect, where materials generate an electric charge in response to the applied mechanical stress, and vice-versa. Therefore, these sensors can function as both actuators and receivers, enabling a dual role that allows them to both emit and detect elastic waves. When SA function as actuators they start vibrating in response to electrical excitation and generate elastic waves in the medium. As these waves propagate through the medium, the receivers capture the propagated waves, which carries information about the medium through which they propagate, a diagram of this process can be seen in Figure 10. The location at which a pair of sensors are placed is therefore important, as the pair can only capture detailed information between and around the pair.

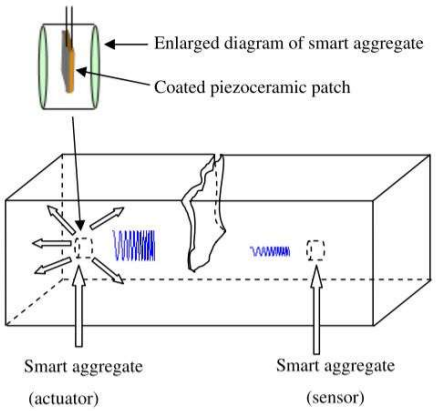


Figure 10, diagram of a SA sensor system, taken from (Song et al., 2008)

The Role of Smart Aggregates in SHM

Unlike traditional sensors, which primarily measure surface-level characteristics, SA operate internally, allowing for a more comprehensive understanding of the concrete structure and condition. In engineering applications, SA are extensively utilized for real-time monitoring of critical infrastructure such as bridges, tunnels, and other large structures (Aggelis & Shiotani, 2007; Cheng et al., 2024). These sensors can be utilized to detect stress (Stähler et al., 2011), temperature (Karaiskos et al., 2015) and the presence of cracks (Karaiskos et al., 2015), providing comprehensive information that helps in proactive maintenance and prolonging the lifespan of structures. Moreover, the SA could provide a NDE approach to monitoring of the structural health of a beam or concrete structure.

UPV is a well-established non-destructive testing technique used to evaluate the quality and integrity of concrete structures. UPV is widely used for quality control in construction, assessing the uniformity of concrete as it relates to cracking to a concrete structure, both in a laboratory and in-situ situation (Camassa et al., 2020). This thesis looks at the application as it contains to crack development, specifically the location and orientation of the crack development.

Whereas, CWI has also been applied to the field of concrete structures, primarily for monitoring the health of concrete in both laboratory and in-situ situations (Niederleithinger et al., 2018; Stähler et al., 2011). Some applications of CWI in concrete structures include monitoring thermal-induced changes (Y. Zhang et al., 2013) and stress-induced change (Planès & Larose, 2013). The technique is highly effective for identifying defects such as micro-cracks (Fröjd & Ulriksen, 2017; Planès & Larose, 2013) and stress variations (Planès & Larose, 2013; Stähler et al., 2011) in concrete. This thesis will try to leverage the high sensitivity and effectiveness of the CWI technique to detect and analyse these defects, such as micro cracks and stress changes, in precast continuous beam elements.

One of the most critical contributions of SA and the CWI method is their ability to detect relative velocity changes within the concrete (Planès & Larose, 2013). This capability surpasses conventional methods by detecting subtle variations that precede visible damage, thereby improving the predictive accuracy to a SHM systems. Chapter 2.2.2 – Data Processing Techniques

2.2.2 – Ultrasonic Pulse Velocity

Originally developed to evaluate the quality and uniformity of construction materials (Karaiskos et al., 2015), UPV has become a vital technique in structural health monitoring. It works by transmitting ultrasonic pulses through a material and measuring the travel time between sender and receiver. The resulting velocity, influenced by the material's density, elasticity, and internal integrity, helps infer structural properties and detect flaws. Higher velocities typically indicate sound material, whereas lower readings may suggest defects such as cracks, voids, or deterioration.

Ultrasonic Pulse Velocity techniques

In the context of UPV testing, various techniques are used to analyse and interpret data to assess the quality and properties of concrete or other materials. Two notable approaches are the Hinkley Pick method (Távora et al., 2021) and the Akaike Information Criterion (AIC) (Pruchnicki & Opieliński, 2021). The Hinkley Pick and AIC methods will be utilized to identify the onset time of the ultrasonic pulse, which will be used to accurately determine the velocity of the pulse through the material. Once the velocity has been determined, it can be compared to the initial velocity thereby producing a cumulative representation of damage development.
In Figure 11, an overlay is presented of the initial portions of the waveforms recorded from beam S10H2D at load steps 0 kN vs 50 kN and 0 kN vs 500 kN, from SA pair SA0 and SA1. A gradual shift in the arrival time of the first wavefronts is observed across the three cases.

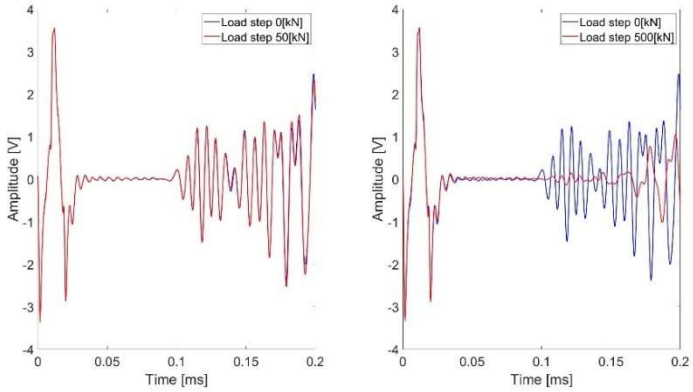


Figure 11, comparison of the initial arrival time of the waves, left load 0 kN (Blue) 50 kN (Red), right load 0kN (Blue) and 550 kN (Red), from Beam S10H2D

Ultrasonic Pulse Velocity in SHM

Within Smart Aggregate–based monitoring in SHM applications, the UPV-derived relative velocity—based on the first-arriving P-wave—functions as a path-dependent indicator of structural integrity along the direct propagation trajectory between sensors. Because the measurement reflects the effective stiffness encountered along this ray path, the indicator is particularly sensitive to discontinuities that intersect or closely approach the transmission path (Karaiskos et al., 2015; Song et al., 2008). This characteristic makes UPV well suited for detecting cracks that interrupt load transfer across a monitored region, as shown in Figure 12 here below.

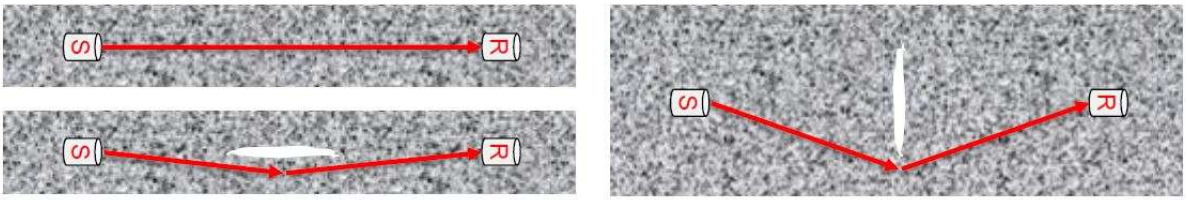


Figure 12, visual representation of the influence of the crack on the wave trajectory (S: source and R: receiver), adapted from(S. Zhang et al., 2022)

The interpretative value of relative velocity is strongly influenced by crack orientation with respect to wave propagation. Cracks oriented approximately perpendicular to the transmission path produce a more pronounced change in travel time due to disruption of stress transfer across the propagation corridor. Conversely, cracks aligned parallel to the wave trajectory exert a reduced influence, as wave propagation remains comparatively uninterrupted.
However, the indicator represents a response along the propagation path and therefore exhibits limited sensitivity to damage located away from the ray path. UPV is thus most effective for identifying and characterising structural disturbances that directly affect the transmission

corridor between Smart Aggregates, while precise localisation of distributed damage requires interpretation across multiple sensor pairs.

2.2.3 – Coda Wave Interferometry

The method known as CWI was first created for interpreting seismic wave data to reveal subsurface geological structures (Planès & Larose, 2013). Recently, CWI has been adapted in NDT and SHM for examining the internal structures of various materials. This technique leverages the sensitivity of coda waves—the later-arriving multiply scattered waves following an initial seismic event—to detect minute changes and anomalies within a material, making it highly effective for monitoring structural integrity and identifying defects such as micro-cracks and stress variations. As the waves pass through the medium, they interact with the medium's internal features, such as heterogeneities and micro-cracks. These interactions cause the waves to undergo multiple reflections and scattering events, altering the path of the waves. As a result of their prolonged travel path, coda waves are particularly sensitive to small changes within the medium, this is due to the accumulations of small changes along the wave path. This sensitivity allows coda waves to provide valuable insights into subtle structural variations and internal conditions that might not be detectable through other methods.

Coda Wave Interferometry Techniques

CWI involves comparing a recorded measurements to an initial reference measurement. By establishing a reference measurement and then analysing new measurements against this reference (Snieder, 2006), CWI can detect and quantify changes in the material's internal structure. These changes will be reflected in the waveform similarities. Some techniques which encompass this method are 'Moving Window Cross-Correlation, 'Dynamic Time Warping' and 'the stretching technique' (Cheng et al., 2024; Zhong & Zhu, 2022).

The stretching technique takes the recorded measurement and stretches or compresses it in the time domain to achieve the best correlation with a reference measurement. In Figure 13, an overlay of the later-arriving coda waves from beam S10H2D at load steps 0 kN vs 50 kN and 0 kN vs 500 kN is shown from SA pair SA0 and SA1. Differences in the time alignment of these waves can be seen, indicating a shift in the time domain due to accumulated internal changes.

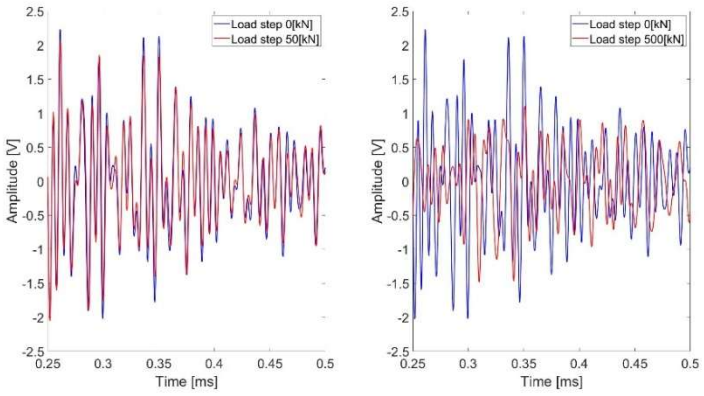
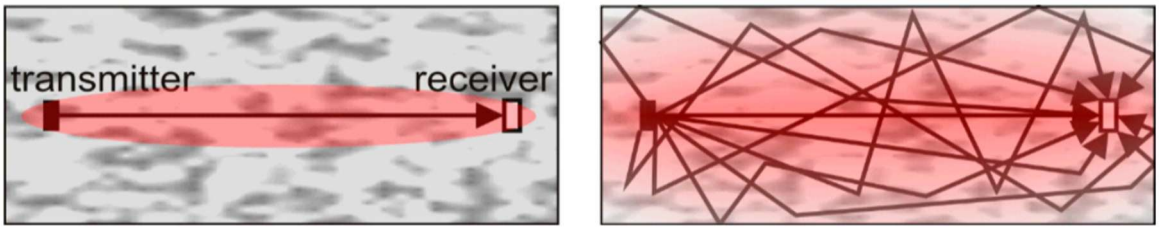


Figure 13, comparison of later arriving waves, left load 0 kN (Blue) 50 kN (Red), right load 0kN (Blue) and 550 kN (Red), from Beam S10H2D

The measurements are sensitive to small changes in the medium and are reflected in changes in the velocity or waveform coherence, this sensitivity can be detected by the technique and be used to indicate the initiation of micro crack and changes in stresses in a concrete beam. Using the stretching technique, one could be able to determine the degree of correlation between the reference and recorded measurements which is called the 'Correlation Coefficient' (CC) and subsequently also determine the relative velocity between the two waveforms. The CC value can be used to indicate the onset of micro-cracks, whereas the relative velocity change could highlight the change in stress in the region of the SA (Planès & Larose, 2013).

Coda Wave Interferometry in SHM

Within SHM applications, the indicators derived from CWI provide a distributed measure of structural evolution within the monitored region. Unlike direct-wave measurements, coda waves sample a large portion of the medium through multiple scattering interactions, resulting in sensitivity to small-scale internal disturbances even when these do not intersect the direct transmission path (Planès & Larose, 2013; Snieder, 2006). This is best seen in Figure 14, where the difference in direct-wave and coda waves is sown, with their monitored region highlighted.



(a) (b)

Figure 14, principle of ultrasonic transmission measurements, propagation paths, and areas of influence (red), direct wave (a), multiple scattering or coda (b). Taken from (Niederleithinger et al., 2018)

This characteristic enables early detection of microstructural changes associated with crack initiation (Planès & Larose, 2013), progressive interface degradation (Cheng et al., 2023; Song et al., 2008), and stress redistribution within the material (Planès & Larose, 2013). In this context, each CWI result represents the incremental difference between two compared waveforms, typically corresponding to consecutive load steps, enabling the detection of subtle changes prior to the accumulation of significant damage. Because the indicator reflects cumulative wavefield perturbations, it provides an area-sensitive response rather than a path-specific response. Consequently, reductions in correlation coefficient represent a global loss of waveform coherence caused by distributed structural modifications, while relative velocity variations reflect small effective stiffness changes within the monitored region.

2.2.4 – Enhanced Imaging Assessment in Structural Health Monitoring

To maximize the benefits of the data acquired by the SA to structural health analyse, effective imaging techniques are essential. Raw ultrasonic data, while valuable, requires further post-processing to extract meaningful patterns and trends a visual manner. One of the more promising imaging techniques is tomography, which reconstructs internal variations within the structure based on wave propagation data.

Tomography is an imaging technique used to reconstruct the internal structure of an object by collecting data from multiple perspectives around the object. This process involves the systematic acquisition of measurements, such as wave velocities or attenuation values, which reflect how waves or particles interact with the material's internal properties (Pereira dos Santos & Guilherme Haach, 2022; Takiguchi, 2019). Using mathematical algorithms, tomography combines this data to create detailed cross-sectional or 3D images of the object's internal features. These images allow researchers and engineers to identify anomalies such as voids or inclusions that could compromise the object's integrity. The method’s versatility and precision have made it invaluable in various fields, including medicine (Zengqiu et al., 2024), geophysics (Takiguchi, 2019), and NDT of structures (Jalinoos et al., 1995; Takiguchi, 2019).

Tomography Techniques

At its core, tomography provides a means to explore an object's internal structure without physically altering it, transforming measured data into visual representations that simplify interpretation and analysis. Various techniques are used to process and translate the collected data into meaningful images, with each method tailored to the specific object and its measurement setup.

An example of an enhanced tomographic reconstruction approach for concrete is presented in *Use of network theory to improve the ultrasonic tomography in concrete (Perlin & Pinto, 2019)*, where wave propagation is modelled within a discretised representation of the inspected domain. As illustrated in Figure 1 of the study and shown in Figure 15, the cross-section of the material is divided into a grid of interconnected elements that collectively represent the possible propagation routes between transmitters and receivers. Instead of assuming a single straight transmission path, the approach considers multiple potential propagation segments through the discretised medium, allowing measured travel-time data to be distributed across the grid according to the material conditions encountered along the wave trajectory.

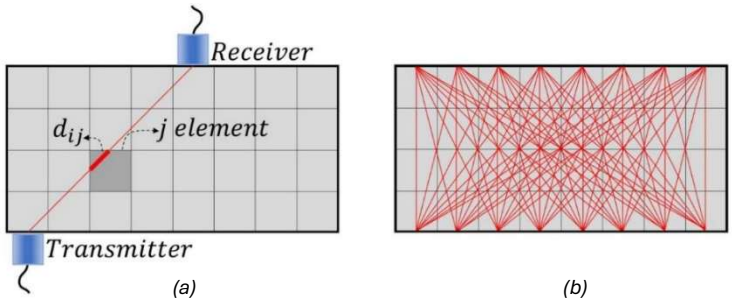


Figure 15, example of an ultrasonic reading, reading i traveling in element j (a) several readings is a section (b), taken from (Perlin & Pinto, 2019)

By combining measurements from many sensor pairs, the reconstruction process produces a spatial distribution of ultrasonic slowness (or velocity), in which each cell represents the local propagation characteristics of the material. Figure 15 conceptually demonstrates this framework by showing how the discretised domain, sensor positions, and propagation routes are combined to form the basis for tomographic reconstruction. Through this representation, variations in wave speed caused by internal heterogeneity or damage are translated into spatially resolved regions within the tomographic image, enabling improved identification and characterization of internal structural anomalies.

Application of Tomography in SHM

Tomographic imaging techniques in SHM commonly rely on velocity-based and attenuation-based methods (Debski & Young, 1999). Velocity-based tomography focuses on measuring the time it takes for ultrasonic waves to travel through different regions of a material. Since wave velocity is affected by material stiffness and internal defects, this method is particularly effective for detecting changes in elasticity, cracks, and stress fields. In contrast, attenuation-based tomography analyses the reduction in wave amplitude as the waves pass through the medium. This approach is sensitive to energy loss due to scattering or absorption, making it useful for identifying areas with high porosity, voids, or microcracking. While both methods aim to reconstruct the internal structure of an object, they differ in sensitivity: velocity-based methods detect mechanical property changes, while attenuation-based techniques highlight energy dissipation patterns.

When applying this reconstruction process to velocity-based tomography, it typically involves setting up and solving a system of equations that links the measured data to the internal structure of an object. These equations are derived from the physical phenomena and behaviour of the material, as mentioned here above. Often, the experimental setup involves acquiring multiple measurements from different locations or under varying conditions, leading to an overdetermined system (Pereira dos Santos & Guilherme Haach, 2022) with more equations than unknowns. In such cases, exact solutions are not possible, so mathematical methods like least squares or iterative reconstruction techniques are employed. These methods provide the best approximation of the internal structure by minimizing errors and progressively refining the solution.

2.3 – Application of Smart Aggregate for Ultrasonic-based Monitoring of Concrete Interface

As highlighted in Chapter 2.1.2, the interface between precast and cast-in-place concrete represents a critical zone governing the composite action of girders made continuous, where structural stiffness and load transfer depend strongly on bond integrity. The bond strength is therefore a crucial aspect in concrete-to-concrete connections, as it governs the effectiveness of shear transfer and composite behaviour; consequently, careful monitoring of this region is essential. Changes in this bond condition—manifested through mechanisms such as microcracking, slip, or delamination—alter wave propagation characteristics within the material and thereby provide a measurable basis for ultrasonic structural health monitoring. Although the literature reviewed in Chapters 2.1 and 2.2 highlights examples of Smart Aggregate–based ultrasonic monitoring and post-processing techniques—namely Ultrasonic Pulse Velocity (UPV) and Coda Wave Interferometry (CWI)—their capability to detect cracking in the interface region remains insufficiently explored.

Building upon this knowledge gap, this thesis investigates Smart Aggregate monitoring in a controlled beam experiment representing a composite precast girder made continuous. Embedded Smart Aggregates enable the acquisition of ultrasonic waveforms at discrete load increments, which are subsequently processed using UPV and CWI techniques to extract multiple ultrasonic indicators. The correlation coefficient and relative velocity change (ε), obtained through CWI, provide high sensitivity to distributed microstructural changes associated with crack initiation and interface degradation. Reductions in CC indicate loss of waveform similarity due to evolving material heterogeneity, while ε reflects effective velocity variations associated with stress redistribution and microcrack development. In contrast, UPV analysis utilises the arrival time of the initial wave to determine relative velocity, enabling path-based detection of stiffness reduction and discontinuities, and supporting localisation of damage along defined propagation paths. To support interpretation, the ultrasonic indicators require appropriate visualisation to relate signal changes to structural behaviour. Tomographic reconstruction is employed to transform multiple sensor pair measurements into spatial fields representing the magnitude and distribution of material changes.

Chapter 3 – Methodology, Data Processing and Analyses Framework for Ultrasonic-Based Monitoring of Composite Precast Girders made Continuous

This chapter outlines the methodology applied throughout the research, beginning with the experimental setup and proceeding through the various stages of data processing and image reconstruction. The aim is to provide a clear understanding of how ultrasonic data was collected, interpreted, and transformed into meaningful representations that reveal structural changes within the concrete beams. Additionally, this chapter presents the mathematical formulas used and explains the processing steps applied to convert raw data into interpretable results. The chapter is structured into four sections to guide the reader through the beam test setup, data processing workflow, imaging process of the data, and finally the criteria used to analyse and interpret the resulting images and graphs.

3.1 – Test Setup and Methodology of the Beam Test

The test and its setup are designed to evaluate the structural response of the beam under controlled loading conditions, ensuring accurate measurement of wave propagation using Smart Aggregate (SA) and enabling the subsequent observation of the internal mechanical behaviour of both the beam and its interface. The key components of the test setup include the beam and its individual elements, the hydraulic jack and the SA in combination with the multiplexer (MUX) system, which records the data. Each component and its role will be discussed in more detail in this chapter.

3.1.1 – Test Setup for the Girder Beam and Data Acquisition System for Smart Aggregates

The test is divided into two main components: the beam test side, which involves applying controlled loading to the concrete beam to simulate structural behaviour, and the data acquisition side, which captures the beam's internal response using SA and a MUX system.

Test Setup and Protocol of the Girder Beam

The test setup includes, from left to right as seen in Figure 16, a half-beam and a full-beam configuration, both connected by the crossbeam in between the two beam elements. The beam is constructed in multiple phases, beginning with the prefabricated T-beam, followed by the crossbeam and topping layer, each poured sequentially. As a result, the initial support conditions evolve throughout the construction process to account for these phases, replicating real-world structural behaviour. This staged approach ensures that differential shrinkage, prestress effects, and interaction between the prefabricated and cast-in-place elements are accurately represented.

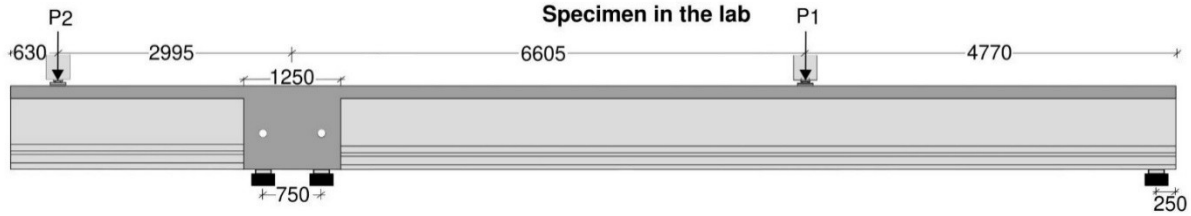


Figure 16, beam test setup, adapted from (Ibrahim et al., 2025)

In the final test setup, the beam is supported at predefined locations while two hydraulic jacks apply load at two points (P1 & P2) along its span, as seen in Figure 16, were the two presses follow a force ratio of $F_2 = 2/3\,F_1$, ensuring a controlled distribution of forces.

For more detailed information regarding the beam configurations, material properties, and specifications please refer to (M. Ibrahim & Yang, 2022). These documents provide comprehensive background and specifications on series test performed by the Structural Mechanics group of the TU Delft. For more detailed information on the load protocol and the overall testing procedure, refer to 3.1.2 – Methodology of the Ultrasonic Test and Data Acquisition.

Data Acquisition System for Smart Aggregates

The SA are positioned around the interface between the prefabricated inverted T-beam and the cast-in-place topping layer, as seen in the figure here above. Each beam contains ten SA, with five embedded in the topping layer and five placed within the inverted T-beam section. The positions of the SA in the T-beam section vary from one beam to another, based on the locations of the hairpin reinforcements. Figure 17 below shows the location of the SA in beam S10H2D. The layouts for all tested beams are provided in Appendix I – Layout of the Smart Aggregates in the Beam.

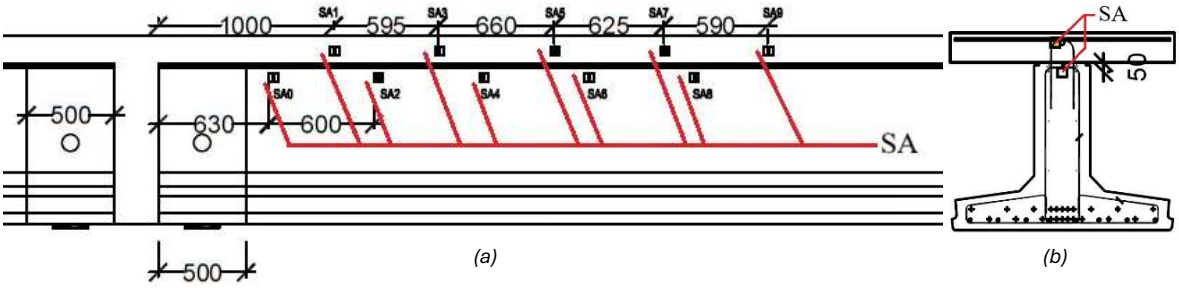


Figure 17, location of the SA in beam S10H2D, longitudinal view (a) and cross-sectional view (b), adapted form (Ibrahim et al., 2025)

The experimental setup employs a MUX system to manage ultrasonic signal transmission and acquisition between the embedded SA. The system consists of a desktop PC, a pulse generator, and two MUX boxes. The MUX allows for systematically switching between the ten available channels, each corresponding to one of the ten SA, ensuring sequential measurement of each SA pair. One SA functions as a transmitter, while the remaining SA serve as receivers. The received signals are routed through the MUX and logged onto the PC for further analysis.
A voltage divider is placed between the pulser and the designated transmitter to adjust voltage levels to within acceptable ranges for the acquisition system. The input voltage from the pulser is reduced to a controlled level of 5V. The system records data at a sampling rate of 3 MHz, which is sufficient to capture the key features of the ultrasonic waveforms. A typical signal received by the SA exhibits a sharp initial arrival followed by scattered energy, with an amplitude in the range of 4 V and a duration of approximately 4 ms. Figure 18 illustrates a systematic representation of the test setup.

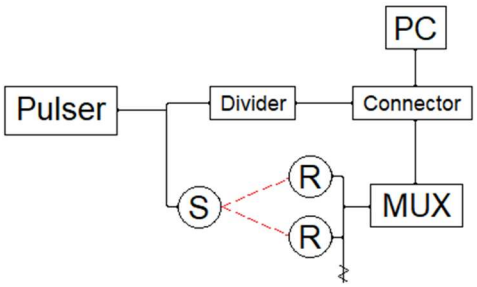


Figure 18, systematic illustration of the Smart Aggregates test setup, with its components; sending SA sensor (S), receiving SA sensor (R)

Some issues may arise during testing or operation of the MUX system, such as electrical interference within the system, which can distort ultrasonic signals and compromise data quality. Additionally, physical problems—such as broken or loosened cables between the sensors and

the MUX—can lead to signal loss, resulting in incomplete or inaccurate datasets. Figure 19 illustrates a representative waveform captured during the testing sequence.

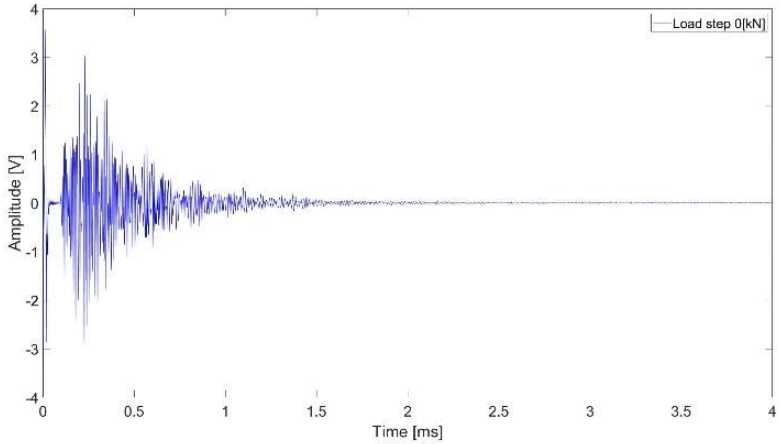


Figure 19, example of a waveform, beam S10H2D SA pair 0-1 at load step 0 kN

3.1.2 – Methodology of the Ultrasonic Test and Data Acquisition

The test methodology follows a structured approach to assess the behaviour of the girder beam under increasing loads. It is divided into three distinct phases: (1) the pre-test phase, (2) the measurement phase, and (3) the post-test phase. During each phase, signals are generated and recorded using the SA and MUX system. The pre-test phase involves recording an initial reference signal before loading begins, establishing a baseline for comparison. The measurement phase consists of generating and recording signals at incrementally increasing load steps, following a strict loading and measurement protocol. Finally, in the post-test phase, once the maximum load level is reached and the beam has been fully unloaded, two additional sets of signals are recorded to assess potential residual changes.

The loading protocol starts with the beam at 0 kN, with initial load increments determined by the load-to-displacement graph. Typically, the load increases in 50 kN steps, while smaller 25 kN increments are used in later stages. However, the final load for each beam is determined based on its structural capacity and observed crack formation.
The measurement protocol dictate that signals are only generated when the load is stable, with the loading procedure temporarily paused to ensure consistent signal acquisition. At each stable load level, the SA operate sequentially, with one SA acting as a transmitter while the remaining nine serve as receivers. This process follows an ordered sequence from SA 0 to SA 9, ensuring comprehensive signal capture. Each set of 90 signals is then systematically logged and categorized.
SA remain inactive during load application and are only activated when the load is stable at a predetermined value. This ensures that only relevant data is collected under controlled conditions. Failure monitoring is crucial throughout the test, as the interface may separate, potentially disconnecting one or more SA. Between load increments, the interface condition is examined to verify the integrity of the SA connections. Visual inspections were routinely performed throughout testing to document any physical changes in the beam or at the interface, as well as to assess the integrity of the connections between the sensors and the MUX system.

3.2 – Post-processing Methods of the Ultrasonic Data

This chapter explains how the raw ultrasonic data acquired from the experimental setup is processed to extract meaningful parameters. It outlines the mathematical principles involved, the step-by-step procedures followed, and the specific challenges encountered during processing for both the UPV and CWI methods. Attention is given to ensuring data consistency,

filtering unreliable measurements, and addressing signal irregularities. These steps are essential to ensure that the final dataset used for visualization is both reliable and physically meaningful.

3.2.1 - Ultrasonic Pulse Velocity Method

The Ultrasonic Pulse Velocity (UPV) method is primarily used to calculate wave velocity by determining the arrival time of ultrasonic waves within a material. Accurately identifying this arrival time is crucial for velocity calculations and subsequent data analysis. In the context of SA signals, the waveform consists of a ballistic part and a coda part. This ballistic component corresponds to the P-wave (primary wave), representing the direct wave traveling through the material, is the most relevant for UPV, as it follows the earliest detectable signal from the first arriving wave.
As seen in an earlier Figure 11, and in the figure here below, the early-arriving portions of the waveform show a gradual shift in arrival time as loading increases, indicating a reduction in ultrasonic pulse velocity.

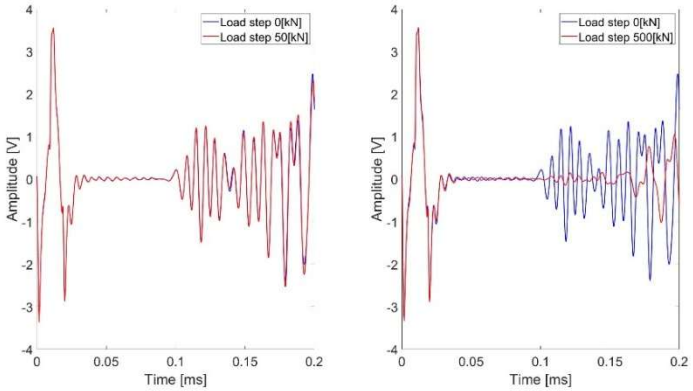


Figure 20, comparison of the initial arrival time of the waves – Beam S10H2D at load steps 0, 50 and 500kN, repeated from Figure 11

The wave velocity ($v_{i,j}$) is calculated using the following equation:

$$v_{i,j} = \frac{d_i}{t_{i,j}} \tag{1}$$

where d_i represents the distance between the sensors in SA pair i (measured in meters), and $t_{i,j}$ denotes the arrival time of the ultrasonic wave for SA pair i at load step j (measured in seconds). The underlining assumption here is that the distance between the sensor pairs remain the same and this is important to accidentally determining the arrival time.

To determine the arrival time of ultrasonic waves, an algorithm employs a two-step method designed for enhanced accuracy and computational efficiency. The process begins with the Hinkley Criterion-based Picker, which identifies significant energy changes in the signal to provide an initial arrival time estimate. This estimate defines a reduced search range centred around the anticipated arrival time. Within this narrowed range, the Akaike Information Criterion (AIC) method is applied to refine the arrival time determination. The AIC method evaluates statistical variances before and after the arrival point, offering superior precision but at a higher computational cost. By integrating the HC method to constrain the search area, the algorithm optimally balances accuracy and efficiency in arrival time estimation. Once the arrival time is determined, wave velocity is calculated based on the measured time and the known sensor distance.

Hinkley Criterion-based Picker

The Hinkley Criterion method detects the onset of a significant change in the signal, making it useful for identifying the arrival of ultrasonic waves in SA signals. It works by recognizing the point where the cumulative energy of the signal diverges from a linear trend, indicating the arrival of the

first significant wave component. The formula for the Hinkley Criterion calculations can be defined as and taken from Távora et al. (2021):

$$S(i) = \sum_{k=0}^{i} \left(x_k^2 - i * \frac{S_n}{n} \right) \tag{2}$$

where $S(i)$ is the cumulative sum of the energy up to sample i, x_k represents the signal samples, and i is the index of the sample for which the energy is being accumulated. S_N denotes the total energy of the selected signal, and n is the number of samples in the selected part of the signal.

Hinkley Criterion algorithm is implemented in the following mathematical steps. First off, designate a starting value and a normalization of the signal is applied to ensure consistent analysis by adjusting its amplitude. Next, compute the cumulative sum of squared signal values, defining the linear trend by calculating the slope of the accumulated energy up to each point. Finally, apply the Hinkley criterion to compare the accumulated energy with the expected linear trend, identifying the minimum deviation from this trend to detect any anomalies.
In the context of the Hinkley Pick-based algorithm, the "minimum deviation from this trend" refers to identifying the point at which the recorded signal first deviates from the baseline noise—this point is interpreted as the arrival time of the initial ultrasonic pulse.

AIC calculation

The AIC works by assessing changes in statistical properties within the signal. It detects the arrival time of an elastic wave by evaluating the variance in the signal before and after each potential pick point. Although commonly used in statistical model selection, the AIC method here identifies the point where the signal changes most significantly, indicating wave arrival. The equation for the AIC calculations can be defined as and take from Cheng et al. (2023):

$$AIC(n) = n * \log_{10}\big(var(S[1,n])\big) + (N - n - 1) * \log_{10}\big(var(S[n+1,N])\big) \tag{3}$$

where $AIC(n)$ is the Akaike Information Criteria value at sample n, var represents the variance of the signal, $S[1,n]$ is the signal segment from the start to sample n, and $S[n+1,N]$ is the signal segment from sample $n+1$ to the end. N is the total number of samples in the signal, and n ranges through all the samples within the selected time window.

AIC algorithm is implemented in the following mathematical steps. First, the signal length must be determined, which represents the total number of samples that will be examined by the AIC algorithm. This length is based on the Hinkley Pick calculation and a fixed value range around it. Once the range has been set, the algorithm performs two main calculations: it determines the variance and then calculates the AIC value. After computing the AIC values for all samples within the selected window, the arrival time is determined by identifying the sample corresponding to the minimum AIC value.

Post-Processing and Evaluation of UPV Results

The UPV algorithm yields a dataset that records the velocity of ultrasonic signals between all SA sensor pairs across every load step of the beam tests. The resulting dataset will be used as the foundation for two analysis techniques—Ray Path visualization and velocity-based Tomography. Since UPV reflects the integrated effect of material changes along the propagation path, the resulting velocity variations represent cumulative damage. Therefore, normalization with respect to the initial load condition is required to isolate relative changes associated with progressive deterioration.
This normalization is defined as the ratio of the current measured velocity to the initial velocity of that sensor pair, recorded at load step 0, at 0 kN. The equation used for this normalization is:

$$\hat{v}_{i,j} = \frac{v_{i,j}}{v_{i,0}} \tag{4}$$

where $\hat{v}_{i,j}$ represents the change velocity of sensor pair i at load step j, $v_{i,j}$is the absolute velocity at that step, and $v_{i,0}$ is the initial velocity for that same pair. This process results in a dimensionless, comparative dataset that highlights changes in wave propagation due to structural transformations.

Once normalized, the relative velocity values can be used to infer the development of cracks, a reduction in relative velocity suggests that the ultrasonic pulse encounters increased resistance or scattering along its travel path, which may indicate cracking, or changes in material stiffness.
A mild decrease may reflect the initiation of damage, while more significant drops in velocity point to more substantial degradation. For the purposes of this analysis, any relative velocity falling below 0.8—meaning a decrease of more than 20% from the original value—is regarded as a critical threshold. When velocities drop beneath this threshold, the data is considered unreliable for further interpretation.
This cutoff is justified by several limitations in the physical and analytical assumptions underlying the UPV technique (Cheng et al., 2023). When a concrete beam undergoes substantial deformation, the geometry of the structure can change to the extent that the distance between SA sensor pairs is no longer constant. Since the UPV method assumes a fixed travel distance between each sensor pair, any deviation from this assumption introduces significant error in the velocity calculation.
Additionally, as cracking progresses, signal quality often degrades due to increased scattering or partial loss of wave transmission. In such conditions, the algorithm may misidentify the wave's initial arrival time or fail to detect a clear signal altogether. These errors are further compounded by potential sensor misalignment or physical disconnection between the sensors and the MUX system.

Addressing Signal Loss and Noise in the UPV method

To ensure the reliability of the dataset, a manual correction step was incorporated into the data processing workflow. During testing, several datasets were found to contain irregularities caused by either signal noise or loss of connection between sensors and the MUX. These issues became more prominent in later loading stages, often due to beam deformation, interface cracking, or disconnection of the cables. Such disruptions led to anomalies in the calculated velocities, primarily due to the Hinkley Pick-based algorithm failing to correctly determine the arrival time of the initial ultrasonic pulse.
In the Figure 21, an example of noise can be seen in the initial phase of the wave of two beam S10H2D and S10H1B with noise. It was unfortunately not possible to eliminate all the effects of the noise; this had the unfortunate effect that only 3 of the data set were useable. These being beams S10H2D, S10H2A and S10H1D.

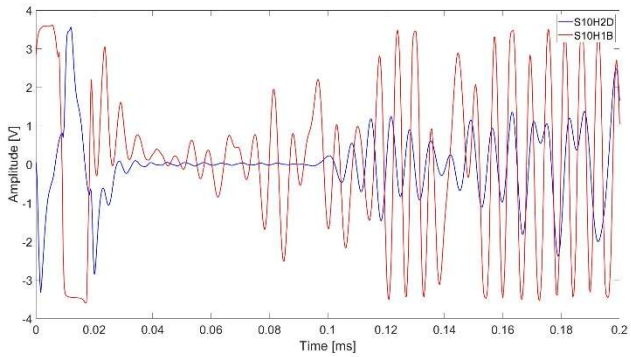


Figure 21, example of noise in the initial phase of the wave, beams S10H2D (Blue) and S10H1B (Red), SA pair 0-1 at load step 0 kN

For the remaining beams, a custom algorithm was developed to manually inspect and adjust these erroneous data points. For each sensor pair, velocity trends were reviewed, and any outliers—typically caused by signal distortion or loss—were flagged. For the affected load steps,

a new arrival time was manually selected and used to recalculate the velocity. However, if the waveform was unusable and the velocity had already dropped below the 80% threshold, a default value of 79% of the initial velocity was assigned to maintain consistency with the lowest validated values observed during earlier analysis. Furthermore, if a sensor was confirmed to have lost physical contact with the MUX, the velocities of all sender-receiver paths involving that sensor were set to <80%, acknowledging a complete failure in transmission.

3.2.2 - Coda Wave Interferometry Method

To evaluate internal changes in structural integrity, Coda Wave Interferometry (CWI) is utilized to monitor alterations in wave propagation. Variations in recorded signals, relative to a baseline measurement, are quantified through the application of a stretching-based comparison. The stretching technique takes the recorded measurement and stretches or compresses it in the time domain to achieve the best correlation with a reference measurement. As seen in an earlier Figure 13, and in the figure here below, the later-arriving coda waves exhibit visible stretching and time-domain distortion.

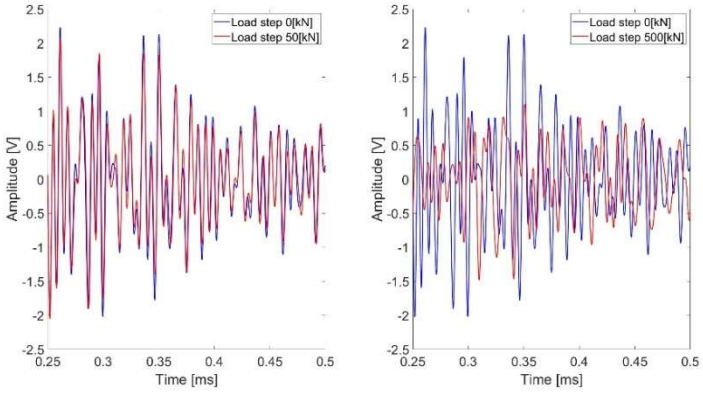


Figure 22, comparison of later arriving waves, beam S10H2D, SA pair 0-1 at load steps 0, 50 and 500 kN, repeated from Figure 13

The measurements by the SA are sensitive to small changes in the medium and are reflected in changes in the wave propagation. This sensitivity can be detected by the technique and be used to indicate the initiation of micro crack and changes in stresses in a concrete beam. Using the stretching technique, one could be able to determine the degree of correlation between the reference and recorded measurements which is called the 'Correlation Coefficient' (CC) and subsequently also determine the relative velocity change (ε) between the two waveforms.
The CC value can be used to indicate the onset of micro-cracks, whereas the relative velocity change could highlight the change in stress in the region of the SA. Recorded signals are compared to a reference measurement, with the relative change quantified using the stretching formula as follows and take from (Cheng et al., 2024):

$$CC(t_c, T, \varepsilon) = \frac{\int_{t_c-T}^{t_c+T} \left(u_{ump}^{(t_c)} u_{per}^{(t_c)}\left(t(1-\varepsilon)\right) \right) dt}{\sqrt{\int_{t_c-T}^{t_c+T} \left[u_{ump}^{(t_c)}\right]^2 dt} \sqrt{\int_{t_c-T}^{t_c+T} \left[u_{per}^{(t_c)}\left(t(1-\varepsilon)\right)\right]^2 dt}} \quad (5)$$

where $CC(t_c, T, \varepsilon)$ is the correlation coefficient calculated over a time window centered at t_c with a duration of $2T$, ε represents the stretching factor, corresponding to the relative change in travel time dt/t. When a uniform velocity change dv occurs within the medium, the stretching factor ε_{max} that maximizes $CC(t_c, T, \varepsilon)$ corresponds to the relative velocity change dv/v.

The calculation algorithm of Formula (5) proceeds in three steps. First, a range of relative velocity changes (ε) is defined, with values spanning from −2e-2 to 5e-3 in steps of 1e-5, and the corresponding time axis is adjusted to align the stretched waveform with the reference. Second, the test waveform is interpolated for each ε to simulate time dilation or compression, after which

the correlation between the stretched waveform and the reference is computed. This comparison is performed within a predefined time window of $2T$ = 200 µs, centred at t_c, where t_c is determined using the Hinkley Criterion method (see Chapter 3.2.1, Hinkley Criterion-based Picker) based on the arrival of later coda waves. Finally, the correlation coefficient is calculated for each ε, and the value that yields the maximum coefficient is selected as the optimal ε, representing the most probable relative velocity change. These steps are repeated for every sensor pair and load step to ensure consistent waveform alignment and accurate monitoring of material behaviour throughout the test.

Post-Processing and Evaluation of CWI Results

The CWI method provides two critical output metrics: the relative change in velocity (ε) and the Correlation Coefficient. These values are calculated directly by the algorithm, and unlike with UPV, no additional normalization or post-processing is required prior to visualization. This is because the method inherently compares waveforms between successive load states, yielding incremental changes between measurements. The outputs can therefore be immediately employed in the Ray-path graphs.

The relative change in velocity serves as a quantitative indicator of structural changes between each loading step. A reduction in ε generally corresponds to a reduction in wave speed, which can signify either increased stress levels or the formation of damage such as cracks. In this study, ε is used primarily to detect the formation of cracks, both at the micro scale. A threshold of ε = -2.0e-4 has been established as the higher bound for reliable interpretation (Cheng et al., 2024). Below this value, the influence of stress becomes significant, making it difficult to separate stress-induced wave speed changes from those caused by cracking. Hence, values under this threshold are considered less dependable.

The CC value provides insight into the signal similarity between load steps and is especially sensitive to microstructural changes such as micro-cracking or scattering. A reduction in CC reflects a deviation in waveform shape, which occurs when the wave encounters a modified internal structure. Based on previous research and experimental validation, a CC value lower than 0.8 is considered too degraded to yield reliable information (Cheng et al., 2024). This threshold serves a dual purpose: first, it helps validate the reliability of the corresponding ε values, and second, it filters out signals that have diverged too significantly from the reference waveform to allow meaningful comparison.

For visualization purposes, ε values are only considered valid if their associated CC value is greater than 0.8. This dependency introduces a limitation during later stages of loading, when stress accumulation and damage formation degrade signal clarity. As loading progresses, both CC and ε become increasingly affected by these stress-induced distortions. Therefore, only the early load steps—typically the first half of the dataset—will be used for final interpretation. This ensures that the visualized results remain within a range where the underlying assumptions of the CWI method hold.

3.3 – Image Reconstruction of the Ultrasonic Indicators

This section details the methodology used to convert the ultrasonic indicators into interpretable graphical images. It describes the Ray Path plotting and tomographic reconstruction techniques, which help visualize possible structural changes. It also covers how the validation process works, and how the invalid or missing data is addressed to ensure the clarity and accuracy of the visual output.

3.3.1 – Ray Path Technique

The Ray Path Technique is a visualization method designed to represent the results of and CWI calculations. By plotting ray paths, this method enables us to easily interpret and identify regions

within the material where changes occur, revealing the location, orientation, and potential propagation of cracks.

Approach to the Ray Path Technique

The process begins with the results obtained from UPV and CWI calculations. These results include change in velocity, relative change in velocity and the 'Correlation Coefficient' value. The data is sorted and grouped by sensor pairs, ensuring all relevant ray paths are captured and organized for visualization. For each sensor pair, an average velocity is calculated prior to plotting. This average is derived from the data recorded when one sensor acts as the sender and the other as the receiver, and vice versa (e.g., the velocity from sensor 0 to 1 and from 1 to 0).
In the next step, the ray path is plotted as a straight line connecting the sensor pairs overlaid on a simplified replica of the beam, with the SA in their corresponding location along the interface, an example of this can be seen in Figure 23.

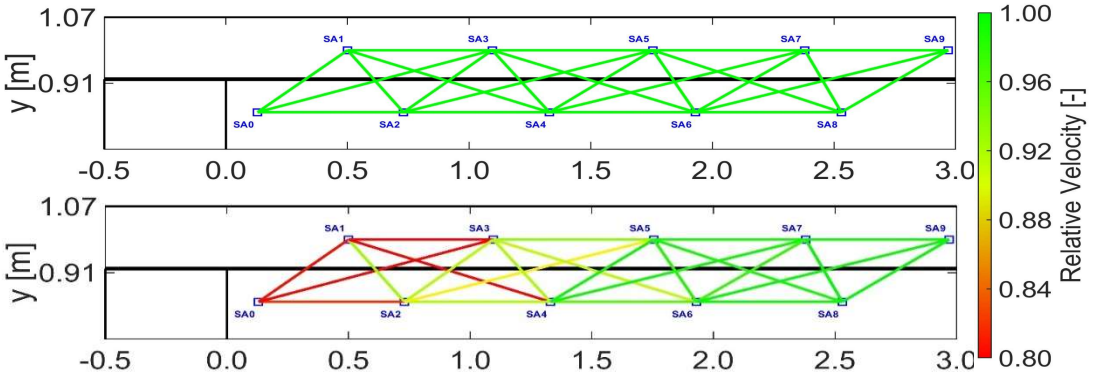


Figure 23, example figure using the Ray Path technique, relative velocity plot of beam S10H2D at load step 0 kN and 500 kN

A key characteristic of the plotted ray paths is the scale and the range of each graph. The range and scale differ over the three results, and these will be determined based on the results to ensure effective visualization. These threshold for each value can be found in their respective chapters here above, also note in these chapter additional notes are covered.
Lastly, a standardized sender-receiver pairing was established based on two criteria: first, the distance between the sender and receiver must be less than 1500 mm; second, no more than six pairings may utilize from a single sensor. These constraints ensure that the most reliable data are prioritized and that the resulting visualizations remain clear and not oversaturated with ray paths. The selected pairings are illustrated in Figure 23 above and in the Appendix I – Layout of the Smart Aggregates in the Beam all the dimensions of the beams and lengths between the SA pairs can be found.

Limitations encountered with the Ray Path Graphs

Most limitations encountered during the graph creation process stem from underlying problems within the UPV or CWI methods themselves. The Ray Path graphs, in essence, simply plot the calculated values from each method into a more suitable and easier-to-understand visual format and therefore do not introduce additional complications. This leads us to the issue of the CWI results, specifically the values that are considered unreliable, and how these should be handled.

During the visualization phase, the results from the CWI calculations present a practical challenge. As discussed in detail in the previous chapter, not all values produced by the CWI method are usable—particularly when the calculated CC value falls below the defined threshold of 0.8. In such cases, the corresponding relative velocity change (ε) is deemed unreliable and is excluded from further analysis.
To account for this, the Ray Path plots are programmed to automatically ignore ε values associated with insufficient CC. This filtering step is integrated into the visualization algorithm, resulting in plots where Ray Paths corresponding to unreliable ε values are effectively 'hidden'. An example of this selective visualization can be found below in Figure 24.

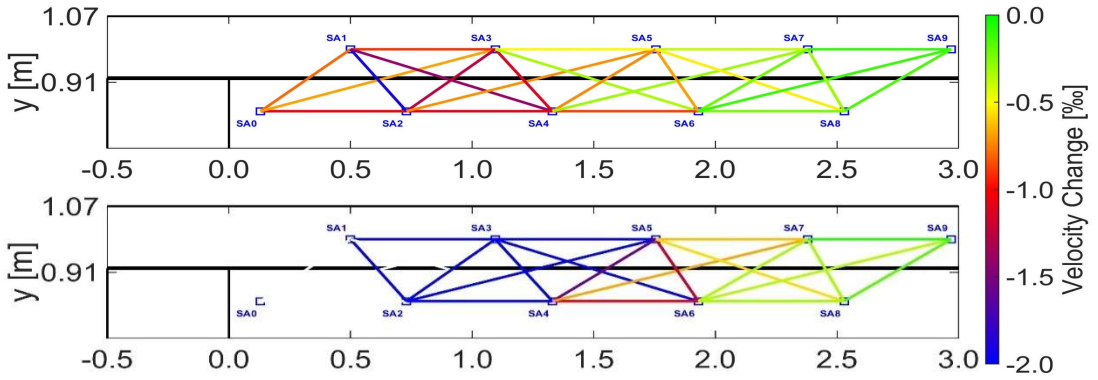


Figure 24, example of hidden Ray Path for SA pairs 0-1 and 0-3, velocity change (ε) plot of beam S10H2D at load 50 and 500 kN

3.3.2 – Tomography Technique

Tomography is an imaging technique used to reconstruct an object's internal structure based on measured wave data. In this thesis, a velocity-based tomography approach is employed to visualize internal changes in the concrete beam under loading. The tomography methodology adopted in this work is based on previously proposed ultrasonic tomography approaches described in Pereira dos Santos & Guilherme Haach (2022) and Perlin & Pinto (2019), which form the conceptual basis for the reconstruction procedure used here.

The process consists of three key steps. First, equations are formulated to represent the relationship between arrival times, field velocities, and interface time interferences. These equations are derived from the geometric configuration of the test setup, the known distances between SA and the arrival times measured across each sensor pair. These equations are then solved using MATLAB's optimization tools—*fmincon*—to estimate unknown variables such as localized field velocities and interface delays. Finally, the tomography graphs are constructed by visualizing spatial variations in field velocity, normalized to baseline values, and relative changes in velocity (ε). Prior to these steps, relevant data—such as travel distances, signal arrival times—must be accurately prepared to ensure precise reconstruction.

Setting up the System of Equations for the Velocity-based Tomography

Tomography fundamentally leverages a system of equations derived from measured data to determine unknown values within a structure. These equations are grounded in established physical principles. In the case of velocity-based tomography, the relationship $v = d/t$ —which links wave velocity (v) to distance (d) and travel time (t) as seen in Equation (1)—is particularly important.

The set of SA sensor pairs and their respective ray paths, as seen in 3.3.1 – Ray Path Technique and Figure 23, form the foundation of the tomographic equations. In the context of UPV-based calculations—used as input data for the tomography model—these equations are then reformulated to incorporate the known travel times between specific SA sensor pairs and the distances their signals travel. The unknowns in this case are the field velocities (v_k) and the interface time interferences dt_l. These are determined based on the layout of the tomography grid and the defined interface sections.

Together, the system of equations and the grid structure form the basis of the tomography analysis, with the key equation expressed as:

$$t_i = \sum_{k \in fields} \frac{d_{i,k}}{v_k} + \sum_{l \in interfaces} dt_l \qquad (6)$$

where t_i represents the measured arrival time for the i SA sensor and its visualized ray path. The term $d_{i,k}$ corresponds to the distance of the ray path of i SA sensor pair that passes through field k, while v_k is the unknown velocity associated with that specific field. Additionally, dt_l refers to the unknown time interference caused by interface section l, which may influence the overall

wave propagation dynamics. In the image here below, you can find how each sensor pair and its ray path intersect a field and or a time interference. This equation is based on the equation (3) from Pereira dos Santos & Guilherme Haach, 2022.

The System of Equations of the Girder Beams

The system of equations can be divided into three sections: the equation themselves, the known and unknown parameters. Like the Ray Path technique the tomography model incorporates the geometrical and physical characteristics of the beam and SA sensor pairs to establish the equations and to derive the known arrival times. Similarly, the unknown parameters—field velocities and interface time interferences—are determined based on the beam's geometry and predefined divisions, as shown in the figure below. This approach ensures consistency in the distance calculations ($d_{i,k}$) and, ultimately, the system of equations, thereby enabling a more universal algorithm for the tomography calculations.

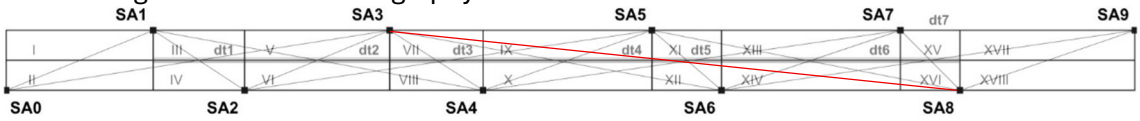


Figure 25, diagram of all the SA Pairs and their ray-paths used for the tomography reconstruction, the velocity fields (I-XVIII) and interface interference line(dt1-dt7), based of beam S10H2D

The tomography model, as seen in Figure 25, divides the beam into 18 distinct fields, each assigned an unknown field velocity (v_k), and incorporates 7 interface time interferences (dt_l) , associated with transitions between the concrete topping layer and the inverted T-beam—bringing the total number of unknowns to 25. From this point forward, the field velocities are denoted as v_I to v_{XVIII}, the interface time interferences as dt_1 to dt_7, the ray path distances as d_1 to d_{68}, and the equations are denoted as eq_1 to eq_{25} with each their respective arrival time denoted as t_1 to t_{68} .

Setting up the Equations in the Tomographic System of Equations of the Girder Beams

In the tomography model, each selected SA sensor pair provides a known arrival time and a corresponding ray path, which traverses specific velocity fields and, in some cases, intersects one interface line. This can be seen in Figure 25, and a zoomed in path 0—1, 0—2 and 0—3 can be seen Figure 26.

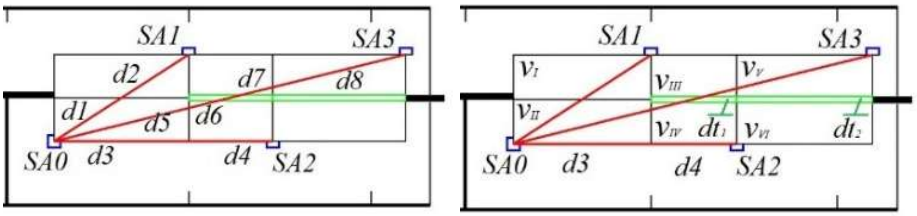


Figure 26, detailed breakdown of the individual components for the equations 0-1, 0-2 and 0-3. Left: Smart Aggregates (SA0 – SA3), their Ray Paths and distance segmented (d_1 - d_{10}). Right: each velocity field (v_I - v_{VI}) and interface interferences (dt_1- dt_2)

These intersections are critical, as they define the components of each unique equation in the tomography system. For every field the ray path crosses, a term is added to the equation representing the travel time through that segment, calculated as the distance of the segment divided by the unknown velocity of that field. Similarly, when the path crosses an interface, an additional delay term is included to account for time interference at that boundary.

As a result, each sensor pair yields a unique equation comprising velocity and interface terms, depending on the specific regions and interfaces intersected by its ray path. This systematic formulation underpins the tomographic reconstruction process.

Examples of such equations (t_1:0-1, t_2:0-2 and t_3:0-3), derived from Equation (6), are shown below and correspond to the ray paths illustrated in Figure 26.

$$SA_0 \leftrightarrow SA_1, eq_1: t_1 = \frac{d_1}{v_I} + \frac{d_2}{v_{II}} \quad (7)$$

$$SA_0 \leftrightarrow SA_2, eq_2: t_2 = \frac{d_3}{v_{II}} + \frac{d_4}{v_{IV}} \quad (8)$$

$$SA_0 \leftrightarrow SA_3, eq_3: t_3 = \frac{d_5}{v_{II}} + \frac{d_6}{v_{IV}} + \frac{d_7}{v_{III}} + \frac{d_8}{v_V} + dt_1 \quad (9)$$

Overview for all the equations can be found in Appendix II.1 – Overview of the System of Equations.

Design Choices on the Fields and Interference Lines in the Tomographic System

The layout of the fields and interface intersections was chosen to balance spatial resolution with the solvability of the tomography system. The total number of unknowns—25 in this case, consisting of 18 field velocities and 7 interface time interferences—dictated the need for a matching number of equations. Based on the original Ray Path technique, only 24 SA sensor pairs are available. To resolve this shortfall, an additional pair—labelled 3—8—was included which can be seen as a red line in Figure 25. While this pair does not meet the standard selection criteria of having a path length below 1500 mm and limiting to no more than six pairs per sensor, it was selected because it intersects multiple velocity fields, providing broader spatial coverage and strengthening the connection between regions. Lastly, note the interface lines are only positioned between SA numbers 1 through 8. The leftmost and rightmost interface locations are not included, as those would be referenced by only one equation each—introducing too many unknowns relative to the available constraints.

Approach to Solving the System of Equations

Even though the number of equations equals the number of unknowns, directly solving them may not produce results that align with real-world conditions. This is where MATLAB's optimization function *fmincon* becomes essential. This tool finds solutions that minimize discrepancies while allowing physical constraints to be incorporated, ensuring realistic values for field velocities and interface time interferences.

The *fmincon* function in MATLAB optimizes a cost function by minimizing the difference between measured and predicted values. It iteratively adjusts unknown parameters within given constraints, refining the solution to best fit the data. The cost function quantifies this difference, ensuring that *fmincon* systematically reduces error while maintaining a physically realistic model. By iteratively refining these values, *fmincon* produces a solution that best aligns with the observed data and given equations, while satisfying the imposed constraints.

Boundary Conditions of the System of Equations

As *fmincon* is highly sensitive to initial input values, these values significantly influence the optimization path and the likelihood of converging to a meaningful solution. Providing them was therefore imperative—not only to guide the algorithm toward physically realistic and stable outcomes, but also to potentially reduce solve time and improve result accuracy. In the end, the optimization process was subject to two key boundary conditions: the initial Velocity Constraints and the Interface Time Constraints.

The calculated field velocities (v_k) must lie within a reasonable range determined by prior UPV and CWI calculations. Therefore, the initial velocities ($v_{k,in}$) for each field will be calculated, using the horizontal SA sensor pairs which travels through a field as base for these initial field velocities, these SA sensor pairs can be seem in the Figure 26.

$$v_{k,j,in} = {}^{d_i}/_{t_{i,j}} \quad (10)$$

where $v_{k,j,in}$ is the initial velocity of field k at load step j, d_i is the distance travelled by the SA sensor pair i, and $t_{i,j}$ is the arrival time of the same sensor pair at the corresponding load step.

The calculated interface time interferences (dt_j) must be non-negative, as interfaces can only slow down wave propagation, not accelerate it. Negative values are physically implausible and would indicate an error in calculation or model assumptions and are therefore excluded from the optimization process.

$$0 < dt_j < 1e^{-4}\ [s] \tag{11}$$

Accuracy Comparison Between Predicted and Observed Parameters

After performing the calculation using the *fmincon* optimization to determine the field velocities or relative velocity change for the tomography graphs, it is crucial to evaluate how accurately the calculated parameters reflect the real-world measurements calculated by the UPV or CWI method. This step is also necessary to access whether the input parameters, system of equations, boundary conditions were correctly chosen. This step would ultimately also help determine the model used for the tomography graphs. Additionally, when dealing with highly constrained systems as this, some level of deviation is expected and must be systematically assessed to ensure that the model's results remain valid and actionable. These constraints, while essential for stable optimization, can introduce discrepancies between predicted model outputs and actual measurements. Therefore, taking both issues in to account two quantitative tools were used to assess the accuracy of the model: Root Mean Square Deviation (RMSD) and Individual Difference (ID).

Comparing the Predicted values to the Observed Values

Both RMSD and ID approaches rely on a comparison between predicted and observed travel time of a given SA pair. The predicted values being those derived from the tomography calculations and the observed values being those derived from the UPV calculations.
The RMSD and the ID metrics serve distinct but complementary roles in evaluating the accuracy of the tomography model. While RMSD provides a global, averaged measure of deviation across all travel-time equations, offering an overall sense of model fidelity, ID focuses on localized discrepancies by expressing the relative error of each individual equation, thereby highlighting spatial variation in prediction accuracy.

Root Mean Square Deviation

The Root Mean Square Deviation is a statistical measure of the average magnitude of error between the predicted and observed values across a dataset (Hodson, 2022). It is especially useful in evaluating the overall performance of inverse models, as it captures both the direction and magnitude of deviation without allowing positive and negative errors to cancel each other out. The RMSD is calculated using the formula:

$$RMSD = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(O_i - P_i)^2} \tag{12}$$

where $RMSD$ represents the magnitude of the error between the predicted and observed values, n represents the total number of equations, or equivalently, the number of sensor pairs used in the tomographic reconstruction. The variable O_i refers to the observed or measured travel time for the $i-th$ equation, while P_i refers to the predicted travel time for that same equation.

In the context of this thesis, RMSD quantifies how far the reconstructed travel times diverge from the measured travel times, on average. A lower RMSD indicates a higher level of agreement between the tomography model and the actual measurements, suggesting that the model parameters (e.g., internal velocities, path reconstructions) are accurately capturing the underlying physical system. Conversely, higher RMSD values may indicate either modelling inaccuracies, over-constrained systems, or limitations in spatial resolution within the tomography setup. The RMSD value obtained from Equation 12 is expressed in seconds and is on

the order of 10^{-4}. Because this magnitude is not directly interpretable without reference to the corresponding travel time, the value is normalised using the formula:

$$NRMSD = \frac{RMSD}{|\bar{O}_i|} * 100\% \tag{13}$$

where $NRMSD$ represents the normalized magnitude of the error between the predicted and observed values in percentage of the absolute mean of the variable O_i, the observed or measured travel time for the $i-th$ equation. With n which represents the total number of equations.

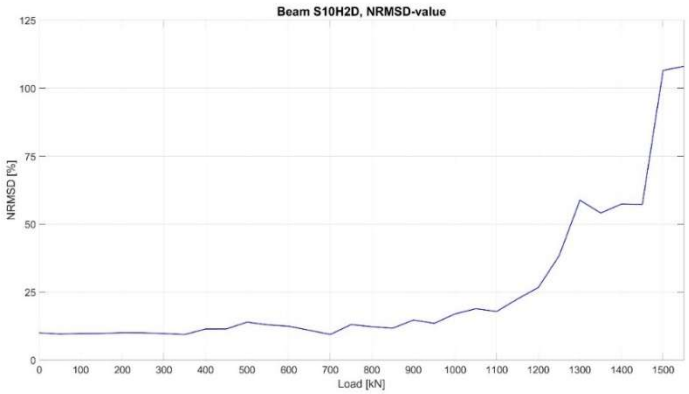


Figure 27, normalised RMSD-value graph of beam S10H2D

Additionally, an increasing RMSD trend observed at later load steps may signal the point at which the tomographic interpretation begins to lose accuracy, as seen in Figure 27, possibly due to accumulating damage, complex wave scattering, or change in the underlying geometry of the model. Identifying this threshold is crucial in understanding the practical limits of the tomographic approach for structural monitoring applications.

Individual Difference

While RMSD provides a holistic view, the Individual Difference (ID) method allows for a more granular, equation-by-equation examination of model accuracy. This metric calculates the percentage error between the predicted and observed travel times for each individual measurement, offering insights into local prediction quality. The ID is defined as:

$$ID = \frac{(P_i - O_i)}{O_i} * 100\% \tag{14}$$

where ID represents the percentage error, P_i denotes the predicted travel time for the $i-th$ equation, and O_i is the observed measured travel time corresponding to that same equation.

High ID values may reveal imbalances in the system of equations, optimization errors, or modelling assumptions, signalling the need for recalibration of parameters or boundary conditions. Elevated ID values in specific regions can also suggest real structural changes, such as cracking or stiffness loss.

Tomography Graph Construction

The process of plotting the tomography graphs closely mirrors that of the Ray Path Technique, following a similar approach in structure, layout, and visualization style. This consistency allows for straightforward cross-comparison between the two methods.

The required data—field velocities and interface time interference—is obtained directly from the processed results discussed in earlier sections. For the velocity-based tomography, normalization is applied to each field's velocity (v_k) and interface time interference (dt_l) using their respective values at the initial load step (see Formula (4)).

All values are sorted by field number and load step, after which the results are plotted using a consistent colour scale, for each value. Colour intensity reflects the magnitude of change, enabling visual identification of regions exhibiting velocity reductions or anomalies. In each tomography graph, both the field velocities and interface values (for UPV) are plotted together, example output is provided in the here below Figure 28 illustration.

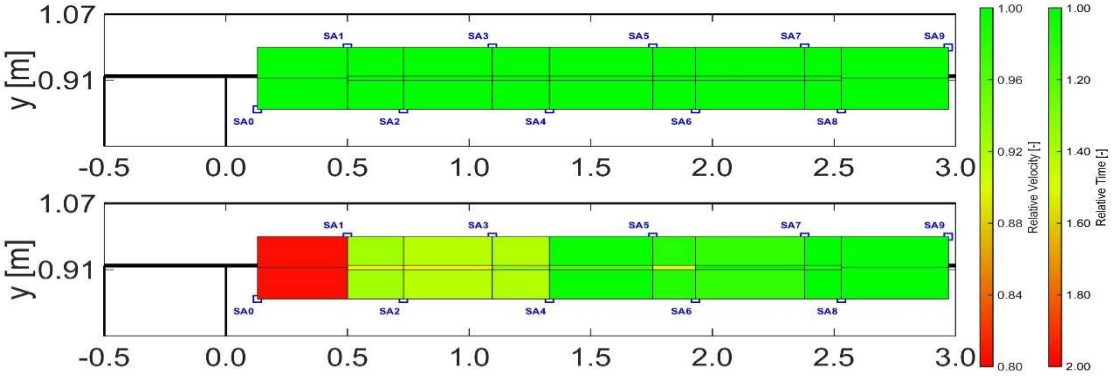


Figure 28, example of a velocity-based Tomography graph, beam S10H2D at load step 0 kN and 500 kN

Limitations of the Tomography Graphs

A fundamental limitation of velocity-based tomography lies in the assumption that the internal geometry of the beam remains unchanged throughout the loading process. In reality, deformation and cracking alter wave travel paths. Therefore, it is essential to use the supporting NRMSD (as discussed in Accuracy Comparison Between Predicted and Observed Parameters) to assess whether a tomography plot remains valid for interpretation. These graphs provide context for evaluating the accuracy of the velocity field assumptions, especially in later load steps where geometric changes are more pronounced.

3.4 – Methodology behind the Comparison, Interpretation, and Validation of the Ultrasonic Graphs

This chapter presents the methodology developed to evaluate the validity and interpretative value of the ultrasonic graphs utilised in this thesis. Building on the ultrasonic data processing and visualisation procedures introduced earlier in Chapter 3, the focus here is on how reliably these indicators reflect internal structural changes, including crack initiation, localisation, and development, with particular attention given to interface delamination. The methodology is applied to three representative beams—S102D, S10H2A, and S10H1D—selected for detailed analysis due to the quality of the acquired test data.

The overarching objective of this chapter is to establish a methodology for assessing whether ultrasonic graphs—and the ultrasonic responses represented within them—provide reliable representations of internal structural changes. The methodology examines how ultrasonic data are visualised and interpreted through these graphs and validates observed trends and anomalies against mechanical behaviour and observations obtained from Digital Image Correlation (DIC).

3.4.1 – Structural Organisation of Ultrasonic Indicators

This section introduces the structural organisation used to support the assessment of ultrasonic indicators. Sub-chapter 3.4.1 focuses on how ultrasonic results are organised and presented, including the categorisation and grouping of SA pairs. These elements define the analytical backbone upon which the interpretation and validation methodology—presented in sub-chapter 3.4.2—is built. Furthermore, this sub-chapter explains how the DIC observations were modified to streamline the analysis process.

Categorisation and Grouping of Smart Aggregate Pairs

The categorisation is based on the spatial positioning of the SA within the beam and is intended to relate ultrasonic responses to expected damage mechanisms, while recognising that individual SA pairs may be influenced by multiple crack types. SA pairs are classified into three categories, based of their location within the beam. Category 1 – Top SA pairs, in which both SA are located in the top layer, are positioned such that they are most likely to capture responses associated with flexural cracking under hogging moments. Category 2 – Bottom SA pairs, in which both SA are located within the inverted T-beam section, are positioned to predominantly capture responses associated with shear-related cracking. Category 3 – Diagonal SA pairs, connecting the top and bottom regions, traverse the interface zone and therefore capture responses associated with interface delamination.

It must be noted that these categories are not mutually exclusive in terms of damage sensitivity, each category can be influenced by different crack types. In UPV measurements, a crack developing outside the primary category's location may still affect a given SA pair if it intersects the direct ray path. In contrast, CWI responses—owing to their reliance on late-arriving, multiply scattered waves—may be influenced by flexural cracking, shear cracking, and interface delamination occurring anywhere within the effective area surrounding the SA pair.

For analytical purposes, SA pairs are further organised into spatial groups, each consisting of four SA and their six corresponding pair combinations. For example, Group 1 consists of SA0, SA1, SA2, and SA3, with pairings 0—1 (Diagonal), 0—2 (Bottom), 0—3 (Diagonal), 1—2 (Diagonal), 1—3 (Top), and 2—3 (Diagonal). This grouping subdivides the beam into regions, enabling comparison of ultrasonic responses across zones subjected to different bending moments and shear demands. An overview of Groups 1 through 4 and their associated SA pairings is provided in Figure 30.

Setting up the DIC graphs

Before the analysis process, the DIC data are sorted and adjusted to better align with the Ray Path and Tomographic graphs. The sorting process consists of identifying the DIC observation with the closest load level to each SA measurement and pairing them accordingly. In Figure 29, an original DIC graph is shown; all DIC observations were provided by M. Ibrahim.

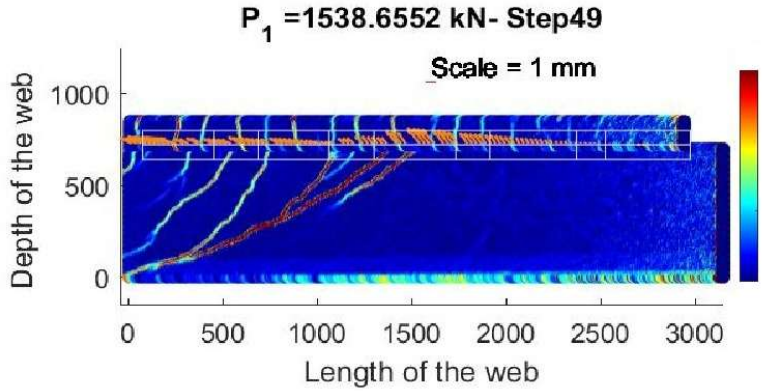


Figure 29, DIC graph with tomography field added, of beam S10H2D at load 1538.65 kN

In Figure 30, the adjusted version is presented, incorporating the field grid used in the tomography analysis, the corresponding SA labels and Ray Paths, and a cropped view displaying the results of one group at a time.

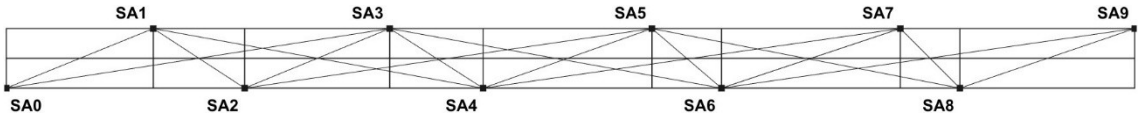
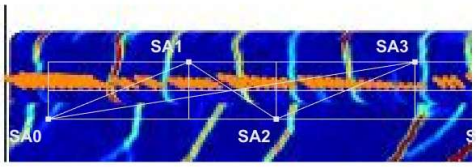
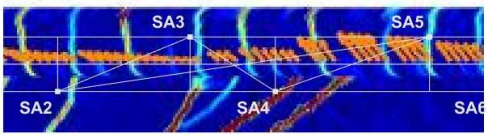


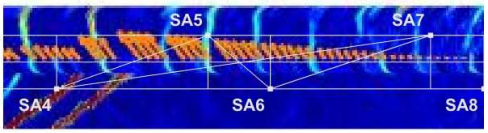
Figure 30, overview of beam S10H2D with SA location (a), DIC observation with adjustments at load 1538 kN, (b) – (e) Group 1 -4



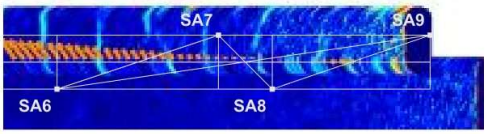
(b)- Group 1, SA0, SA1, SA2 and SA3



(c)- Group2, SA2, SA3, SA4 and SA5



(d)- Group 3, SA4, SA5, SA6 and SA7



(e)- Group4, SA6, SA7, SA8 and SA9

These adjustments are introduced in the analysis process to help the viewer place the SA and their ray paths within the DIC observations. Additionally, they are intended to streamline the terminology used to describe crack locations and to facilitate a more consistent comparison between the DIC observations and the ultrasonic results. However, it must be noted that these additions and adjustments are not entirely exact, as the DIC observations had to be manually reshaped to fit the grid pattern. Some errors may occur and must be noted when they arise.

3.4.2 – Methodology for the Sequential Analysis and Validation of the Ultrasonic Results

The methodology presented in the following chapters consists of three principal stages. However, prior to the sequential analysis, each beam—and each corresponding group along the beam—is first examined using the modified DIC observations, where each crack is identified and recorded by load stage, crack type, and location. These group-based analyses are not included in the main results presented in Chapters 4 and 5 but serve as a reference for subsequent analysis; they are provided in Appendix IV – Group Analysis.

The first stage is the area-based analysis, where the response behaviour within each spatial group is examined using CWI overview graphs. The second stage involves the analysis of unified ultrasonic graphs to evaluate the evolution of ultrasonic indicators along individual SA pairs. The third stage assesses tomographic reconstruction results to evaluate spatial consistency and reconstruction reliability. At each stage, points of interest are identified and compared with experimental observations to assess their physical plausibility.

Points of interest (POI) are defined using two criteria. The first criterion is the exceedance or approach of threshold values of the respective ultrasonic indicator. The second criterion is the occurrence of local minima or pronounced deviations within a defined load interval, hereafter referred to as valleys. These POIs form the basis for subsequent comparison with observed crack behaviour.

For consistency across all ultrasonic representations, unified graphical scaling is applied. The horizontal axis represents load levels up to 750 kN, corresponding to the loading range for which CWI data are available and ensuring comparability between CWI-based and UPV-based responses. The vertical axis range is expanded to approximately twice the magnitude associated

with reference threshold values, reported in previous studies and chapters. This scaling is adopted to enable examination of trend development and to assess whether established critical thresholds are applicable to the experimental results obtained in this thesis. This unified scaling framework is applied to CWI overview graphs, unified ultrasonic graphs, and tomographic overview representations. The only exception concerns the tomography overview graphs, for which the load range is extended where justified by reconstruction reliability, as determined through evaluation of the NRMSD metric.

CWI Overview Graphs for Area-Based Assessment

CWI overview graphs constitute the initial analytical step and support area-based assessment within each spatial SA group. These graphs present the evolution of a single CWI indicator—either ε or CC—for all SA pairs within a defined beam region as a function of load step. Their purpose is to identify spatially distributed ultrasonic response behaviour that may reflect regional cracking activity or interface disturbance, with particular focus on whether identified points of interest correspond to the onset of cracks observed in the DIC measurements.

Unified Ultrasonic Graphs for SA Pair Assessment

Following identification of region-based response behaviour, unified ultrasonic graphs are used to evaluate indicator evolution along individual SA propagation paths. These graphs combine the CWI-based indicators (ε and CC) with the UPV-based relative velocity for a single SA pair into a unified representation plotted against applied load steps.
The primary objective of this representation is to provide a compact overview of ultrasonic indicators development along a specific propagation path. Within each unified graph, POI identified from indicator evolution are compared with corresponding DIC observations, to assess whether ultrasonic indicators reflect observed crack behaviour within the structural region associated with the propagation path.

Tomography reconstruction Validation and Assessment

Tomographic reconstruction assessment is performed to evaluate the spatial consistency of ultrasonic response interpretation and to verify reconstruction reliability. The assessment is conducted in two stages. First, reconstruction quality is evaluated using the NRMSD metric introduced in Chapter 3.3.2.3. Second, group-based overview graphs are constructed for tomographic indicators, including field velocity and interface time interference. These overview graphs follow the same comparative structure as the CWI overview graphs, presenting indicator evolution across SA pairs within defined beam regions.
The tomographic results are subsequently examined to determine whether spatial variations in reconstructed velocity fields and interface-related indicators correspond to experimentally observed crack development and interface behaviour within the associated structural regions.
Emphasis is placed on the assessment of the interface time interference indicator, as this parameter is unique to the tomographic analysis. A more extensive evaluation is conducted for this indicator due to the absence of established reference thresholds in the literature; provisional threshold values are defined at 1.5, 2.0, 2.5, and 3.0.

3.4.3 – Contextual Evaluation of Ultrasonic Indicators in Ultrasonic Graphs

Following sequential validation against experimental observations from DIC, the ultrasonic indicators are evaluated with respect to their interpretative capability within the ultrasonic graphs. The assessment examines how effectively the graphs convey the information contained in the ultrasonic indicators through both Ray Path representations and tomographic reconstructions. The evaluation is structured around three aspects of crack and interface behaviour: early detection of anomalies, localisation of cracking within the structural region, and representation of crack magnitude and progression. Observations and representative examples

from the experimental results are used to systematically review and discuss these aspects of performance.
Following this evaluation, a combined interpretation approach is considered, in which CWI-based indicators are primarily assessed for their sensitivity to early-stage anomalies, while UPV-derived relative velocity is evaluated for its capability in spatial localisation of developed cracking.

3.5 – Synthesis of the Ultrasonic Monitoring Methodology of the Composite Precast Girders made Continuous

The experimental methodology is centred around a controlled beam test representing a precast girder made continuous through a cast-in-place connection. Smart Aggregate sensors were embedded within the concrete to enable ultrasonic measurements at predefined area on the beam and around the interface region. Ultrasonic signals were recorded at discrete load increments, allowing the evolution of the waveforms to be tracked as the structural behaviour progressed from crack initiation to further damage development. This measurement strategy produced a structured dataset consisting of repeated ultrasonic measurements for each sensor pair across successive load steps, forming the basis for subsequent signal processing and interpretation.

Two complementary ultrasonic processing methods were applied to extract meaningful indicators from the recorded waveforms. The UPV method evaluates the arrival time of the first wave packet travelling between sensor pairs, providing a direct measure of effective wave velocity along the propagation path. Accurate arrival time determination was achieved using the Hinkley picker, enabling consistent identification of the first significant signal arrival.
In parallel, CWI was applied to the later portions of the ultrasonic signals. Through waveform stretching and correlation analysis, two indicators were obtained: the correlation coefficient, which quantifies waveform similarity relative to a reference state and reflects changes in material heterogeneity, and the relative velocity change (ε), which captures small velocity variations associated with stress redistribution and microstructural damage.

To facilitate interpretation, two visualisation approaches were employed. In the Ray Path representation, ultrasonic indicators are mapped directly onto the propagation paths between sensor pairs, enabling disturbances to be related to specific locations and orientations within the beam. In the tomographic representation, multiple sensor-pair measurements are redistributed across a spatial grid to reconstruct field velocity distributions and the interface time interference indicator (dt). The field velocity provides a spatial overview of velocity variations within the monitored region, while dt represents localised time delays and is evaluated for its potential to indicate interface disturbances such as delamination.

To support interpretation and validation, the results were evaluated through multiple analysis approaches, including area-based analysis, ray-path analysis, and assessment of the tomographic reconstruction, the latter addressing both accuracy and reconstruction effects. These analyses enable the consistency between ultrasonic indicators and observed structural behaviour to be examined across different representations. Subsequently, both graph types are evaluated with respect to how each ultrasonic indicator is represented, incorporating findings from the evaluation analyses, and assessing their ability to convey information on early detection, crack localisation, and damage progression. Finally, a combined assessment is performed to evaluate how effectively the graphs, through integration of CWI and UPV indicators, support the detection and localisation of cracking.

Chapter 4 – Assessment and Validations of the Ultrasonic-based Indicators

This chapter presents the assessment and review of the ultrasonic indicators obtained from the experimental campaign. The evaluation follows the methodology described in Sub-chapter 3.4, where the procedures for processing the ultrasonic measurements and deriving the corresponding indicators were introduced. In this chapter, the results of these indicators are analysed in relation to the applied load levels and compared with the available Digital Image Correlation (DIC) observations to assess their ability to detect and track internal damage. The chapter is structured by analysis type and subsequently by beam, guiding the reader through the interpretation of the ultrasonic indicators and their correspondence with the observed structural behaviour.

The analysis focuses on beams S10H2D, S10H2A, and S10H1D. For each beam, different Smart Aggregate (SA) pairing groups are examined depending on the availability and reliability of the post-processed ultrasonic indicators. Where applicable, the ultrasonic indicators are compared with the corresponding DIC observations to support the interpretation of the results.
Since DIC images are not available for every load step, these instances are explicitly noted in the discussion. A complete overview of the DIC observations and ultrasonic indicator results for all load levels and groups is provided in Appendix IV – Group Analysis and Appendix V – Ultrasonic Graphs Analysis.
The objective of this chapter is to examine the Points of Interest (POI) identified in the overview graphs and compare them with the corresponding DIC observations to evaluate how effectively the ultrasonic indicators capture crack development. Three types of damage mechanisms are considered: flexural cracking (highlighted in red), shear cracking (highlighted in yellow), and delamination (highlighted in green). The results are assessed with respect to three key aspects: early detection capability, localization of the damage, and the magnitude of the indicator response. Each analysis first presents a description comparing the identified POI with the corresponding DIC observations, followed by additional comments where notable behaviour in the indicators is observed.

4.1 – Area Analysis of the Ultrasonic Indicators

In this section, the ε and CC indicators are examined for each beam at the group level. For beam S10H2D, both ε and CC are evaluated for Groups 1, 2 and 3. For beam S10H2A, ε is examined for Groups 1 and 2, while CC is evaluated for Group 1 only. For beam S10H1D, both ε and CC are examined for Groups 1, 2 and 3. For each group, indicator evolution across load levels is presented together with the corresponding DIC observations where available.

4.1.1 – Beam S10H2D

For Group 1, ε-based POI occur at 100 kN, 350 kN and 500 kN. At 100 kN, all SA pairs exhibit a slight reduction in ε relative to the preceding load step, while no cracking is observed in the reference observation. At 350 kN, ε-values again show a noticeable reduction across all SA pairs, with pair 2—3 exhibiting the smallest change. The corresponding reference observation shows two flexural cracks; one located outside the monitored area and one adjacent to SA1. At 500 kN, ε-values again show a reduction across the monitored region; no reference observation is available for this load level.

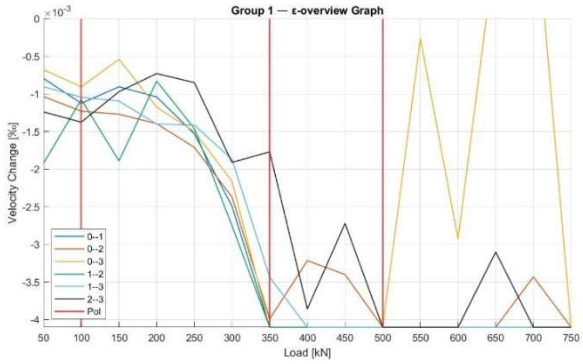


Figure 31, Beam S10H2D Group 1 ε-overview Graph

The first ε-based point of interest occurs at 350 kN, which coincides with the earliest appearance of a crack adjacent to SA1. An earlier crack outside the monitored region is observed at 250 kN.
The SA pair exhibiting the smallest ε-response at 350 kN is pair 2—3, which is also the pair located furthest from the observed cracks. This suggests that the spatial response of ε is influenced by the proximity of the propagation path to the crack location.
The response observed at 100 kN does not appear to represent a threshold level, as no cracking is observed. At 300 kN, ε-values approach the threshold level across all SA pairs without visible cracking within the monitored area, while at 350 kN the indicator exceeds the threshold, coinciding with the first crack within the monitored region.

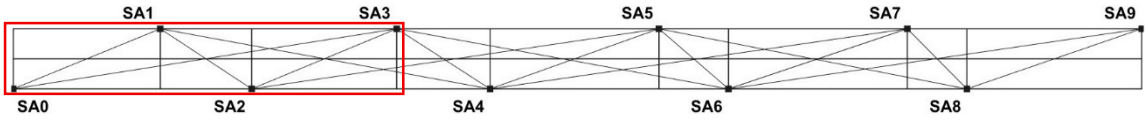
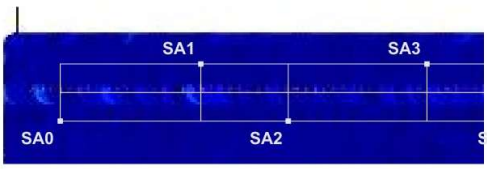
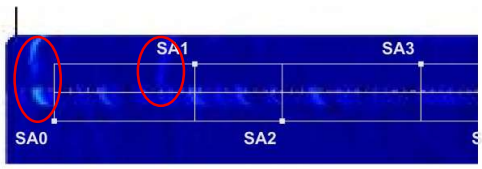


Figure 32, Beam S10H2D, (a) Group location on the beam, (b)-(c) DIC observations



(b)- 100[kN]



(c)- 350[kN]

For Group 2, ε-based POI occur at 100 kN, 450 kN, 550 kN and 650 kN. At 100 kN, all ε-values show a slight reduction relative to the preceding load step, while no visible cracking is observed. At 450 kN, ε-values again show a noticeable reduction across all SA pairs, with pairs 2—5 and 4—5 exhibiting the smallest change; the reference observation shows a flexural crack located left of SA3. At 550 kN, a pronounced ε-response is observed, most notably for pair 2—3; the reference observation indicates clear delamination without additional cracking. At 650 kN, ε-values again exhibit a noticeable reduction compared with the previous load step.

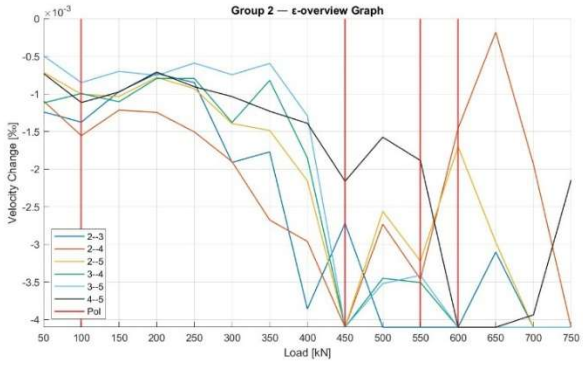


Figure 33, Beam S10H2D Group 2 ε-overview Graph

A flexural crack near SA3 first appears at approximately 400 kN and becomes clearly visible at 450 kN, which coincides with the first ε-based point of interest. At 450 kN, the strongest ε-responses are observed for SA pairs 2—3, 2—4, 3—4, and 3—5, which correspond to the sensor paths located closest to the observed crack near SA3

Before the POI of 450 kN, ε-values exceed the threshold for all SA pairs despite the presence of only a single visible crack. This suggests that additional structural changes or stress redistribution within the beam may influence the ε response beyond the immediate crack location.

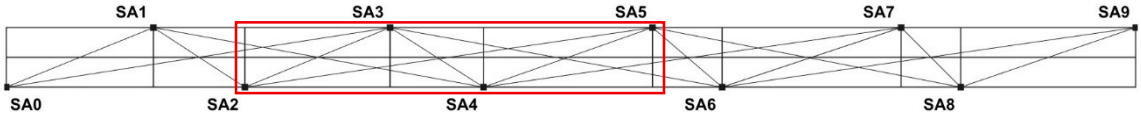
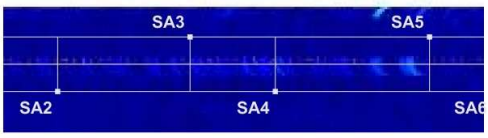
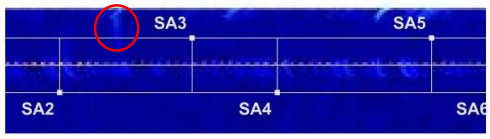


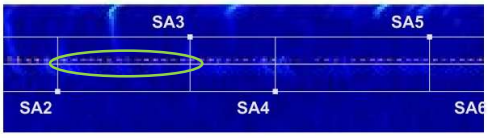
Figure 34, Beam S10H2D, (a) Group location on the beam, (b)-(e) DIC observations



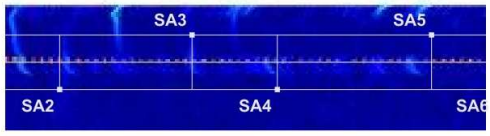
(b)- 100[kN]



(c)- 450[kN]



(d)- 550[kN]



(e)- 600[kN]

For Group 3, ε exhibits only minor responses across the monitored load levels. At 100 kN, 300 kN and 450 kN, slight reductions are observed for selected SA pairs while no cracking is visible. At 600 kN, ε again shows a minor response for most pairs, while pair 4—5 exhibits a pronounced reduction; the reference observation shows small cracks located outside the monitored area.

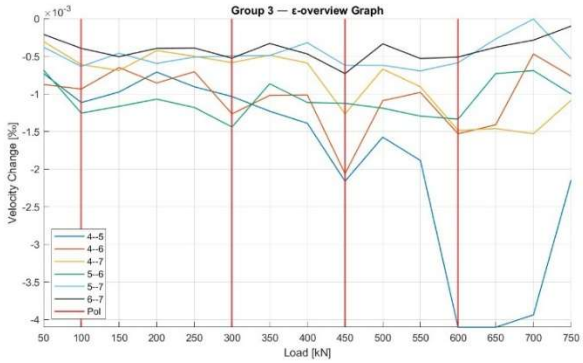


Figure 35, Beam S10H2D Group 3 ε-overview Graph

The POI's observed at 100 kN and 300 kN remain below the defined threshold level and no visible anomalies are recorded in the reference observations at these load steps.

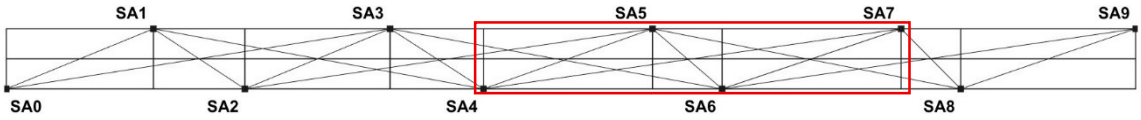
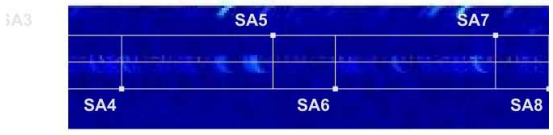
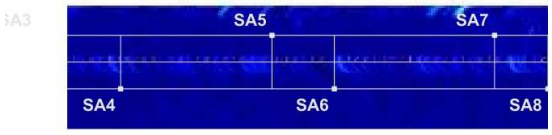


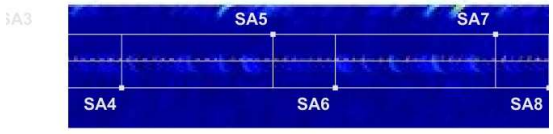
Figure 36, Beam S10H2D, (a) Group location on the beam, (b)-(e) DIC observations



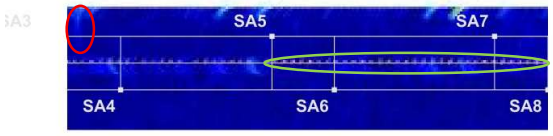
(b)- 100[kN]



(c)- 300[kN]



(d)- 450[kN]



(e)- 600[kN]

For Group 1, CC-based POI occur at 350 kN, 500 kN, 600 kN and 650 kN. At 350 kN, CC-values show a clear reduction across all SA pairs, most notably for pairs 0—3 and 0—1. The reference observation shows two flexural cracks, one outside the monitored area and one adjacent to SA1. At 500 kN, CC again shows a noticeable reduction, most notably for pair 0—3; no reference observation is available. At 600 kN, CC-values again show a reduction while no additional cracking is observed. At 650 kN, CC again exhibits a pronounced response, and a new shear crack is observed near SA2.

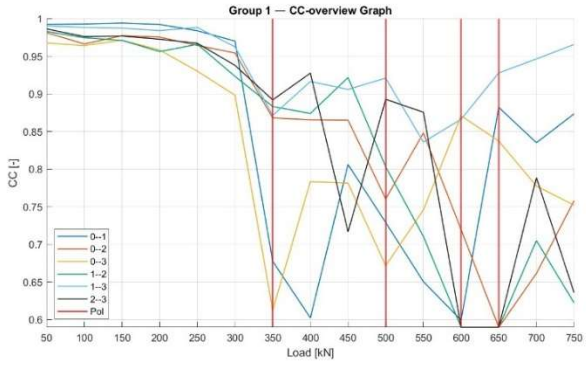


Figure 37, Beam S10H2D Group 1 CC-overview Graph

The POI at 350 kN coincides with the first appearance of a flexural crack adjacent to SA1, while another crack is observed outside the monitored region. At this load step, SA pairs 0—1 and 0—3 exhibit the strongest CC reductions and are also located close to the crack near SA1, while pair 1—3 does not show one of the strongest responses. Similarly, at the POI of 650 kN, SA pairs 0—2, 2—3 and 1—2 exhibit the largest CC reductions and are located closest to the newly observed shear crack near SA2.

At the POI of 350 kN, the SA pairs with the strongest responses fall below the threshold value, while the remaining pairs approach the threshold with CC values around 0.85. After this load step, CC values remain close to the threshold while additional cracks appear at nearly every subsequent load step.

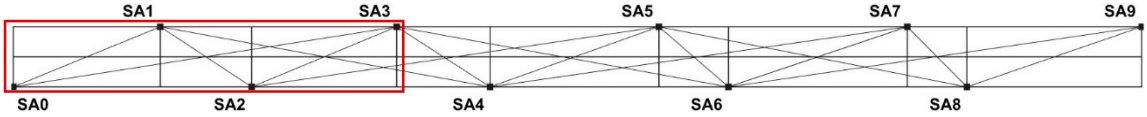
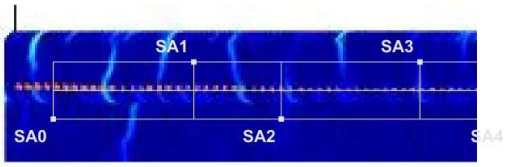
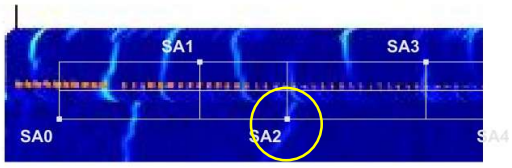


Figure 38, Beam S10H2D, (a) Group location on the beam, (b)-(d) DIC observations

(b)- 350[kN]



(c)- 600[kN]



(d)- 650[kN]

For Group 2, CC-based POI occur at 450 kN and 600 kN. At 450 kN, CC-values show a reduction across all SA pairs, most notably for pairs 3—4, 3—5 and 2—3; the reference observation shows a flexural crack located left of SA3. At 600 kN, CC again shows a reduction without additional cracking.

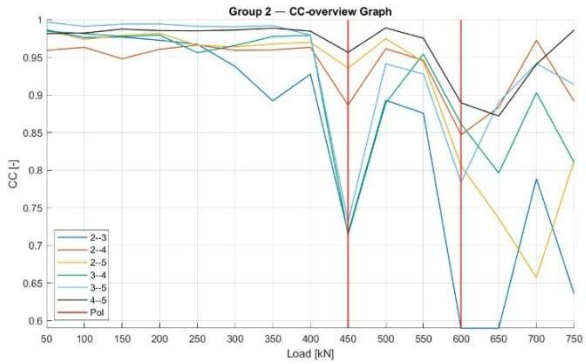


Figure 39, Beam S10H2D Group 2 CC-overview Graph

Compared with Group 1, the CC responses in Group 2 appear less chaotic. This could correlate with less crack development within the Group 2 monitoring region, as observed in the DIC observations.

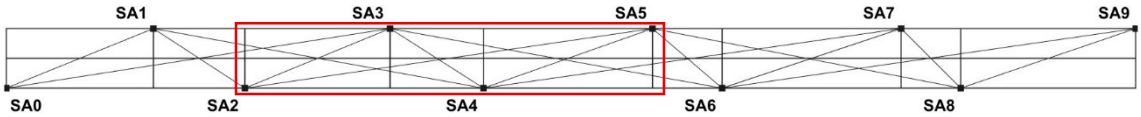
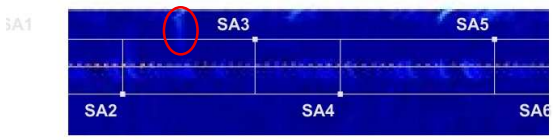
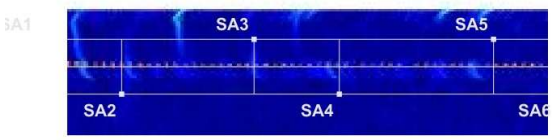


Figure 40, Beam S10H2D, (a) Group location on the beam, (b)-(c) DIC observations



(b)- 450[kN]



(c)- 600[kN]

4.1.2 – Beam S10H2A

For Group 1, ε-based POI occur at 450 kN and 600 kN. At 450 kN, ε-values show a reduction for selected SA pairs, however no corresponding reference observation exist at this load. At 600 kN, ε-values show a reduction across all SA pairs, with pairs 2—3 and 0—2 exhibiting the smallest change; the reference observation shows increased delamination and a new flexural crack near SA1.

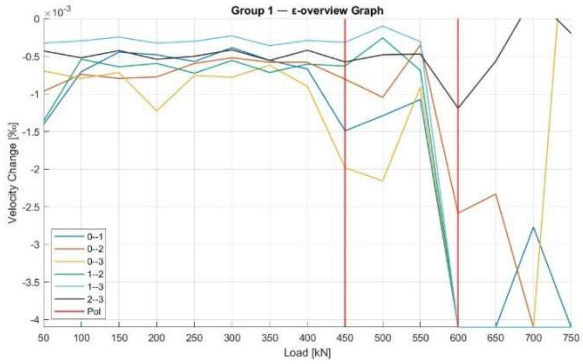


Figure 41, Beam S10H2A Group 1 ε-overview Graph

At the POI of 600 kN, the first clear appearance of delamination is observed. No clear indication is observed that the delamination location can be identified in the ε responses.

Figure 42, Beam S10H2A, (a) Group location on the beam, (b) DIC observations

(b)- 600[kN]

For Group 2, ε-values show a reduction at 600 kN across all SA pairs, most notably for pairs 2—3, 2—5 and 2—4. The reference observation shows increased delamination without the appearance of new cracks.

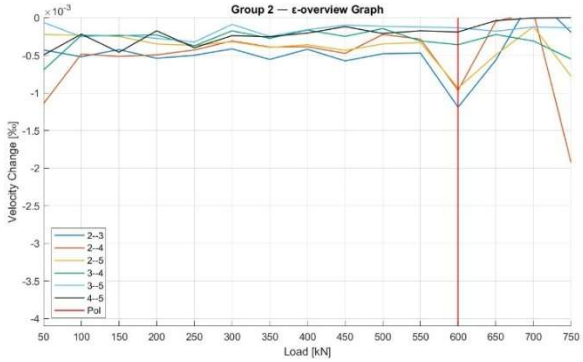


Figure 43, Beam S10H2A Group 2 ε-overview Graph

The observed delamination did not first appear at this load. Additionally, the ε-values of the whole group never goes below -1.5e-4, corresponding to the lake of crack development in this monitoring area.

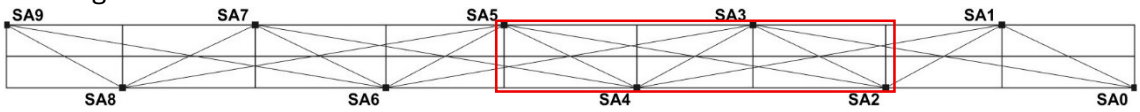
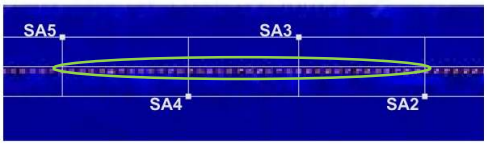


Figure 44, Beam S10H2A, (a) Group location on the beam, (b) DIC observations



(b)- 600[kN]

For Group 1, CC-based POI occur at 500 kN, 600 kN and 650 kN. At 500 kN, CC shows a reduction for pair 0—1 without a corresponding reference observation. At 600 kN, CC-values show a reduction across all SA pairs, most notably for pairs 1—3 and 0—1 and except for 2—3; the reference observation shows flexural cracking to the right of SA1. At 650 kN, CC again exhibits a pronounced response for most pairs, except for 2—3, while a new shear and flexural cracking is observed near SA1.

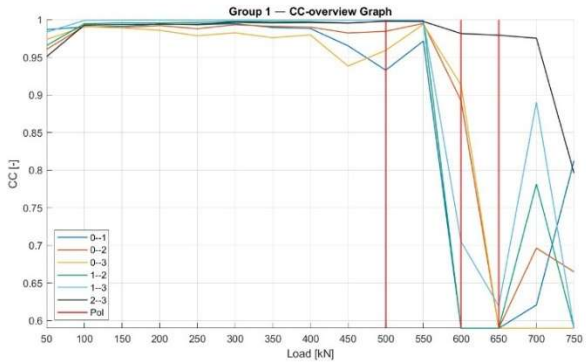
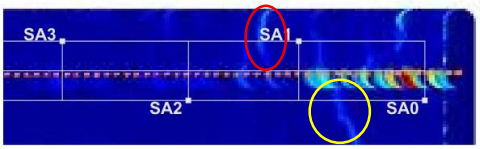


Figure 45, Beam S10H2A Group 1 CC-overview Graph

The POI at 600 kN corresponds with the first clear observation of delamination, the CC responses do not show a clear spatial relationship with the observed delamination. A second valley at 650 kN could coincides with the first appearance of a shear crack near SA0. The SA pairs exhibiting the strongest reductions are not directly aligned with the shear crack location.

Figure 46, Beam S10H2A, (a) Group location on the beam, (b)-(c) DIC observations

(b)- 600[kN]



(c)- 650[kN]

4.1.3 – Beam S10H1D

For Group 1, ε-based POI occur at 200 kN and 350 kN. At 200 kN, ε-values show a reduction across all SA pairs without visible cracking. At 350 kN, ε again shows a noticeable response and the reference observation shows early flexural cracking left of SA1.

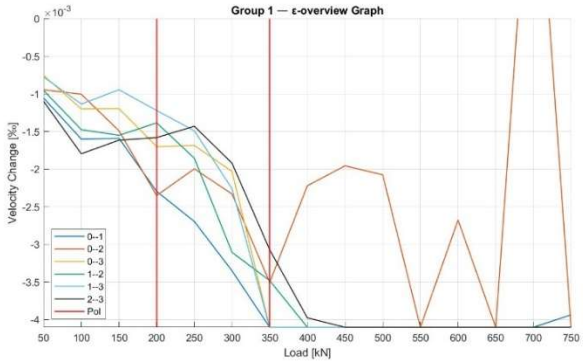


Figure 47, Beam S10H1D Group 1 ε-overview Graph

At 200 kN, ε-values for all SA pairs are close to the threshold value while no cracks are observed. At 350 kN, ε-values for all SA pairs fall below the threshold while cracks are still not clearly visible in the observations. This may indicate that additional factors influence the ε response.

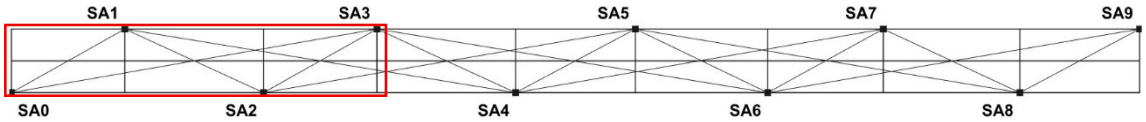
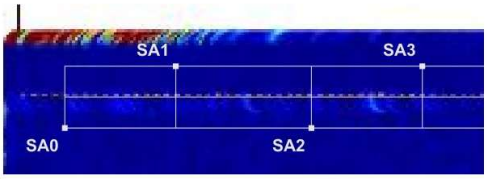
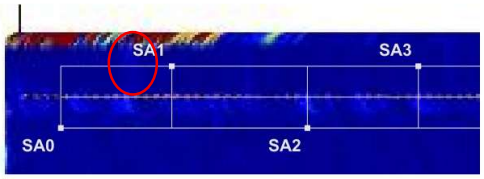


Figure 48, Beam S10H1D, (a) Group location on the beam, (b)-(c) DIC observations



(b)- 200[kN]



(c)- 350[kN]

For Group 2, ε-based POI occur at 100 kN and 500 kN. At 100 kN, ε-values show a reduction across the monitored region without visible cracking. At 500 kN, ε again shows a reduction, with pair 4—5 exhibiting the smallest change; the reference observation shows a flexural crack located left of SA3.

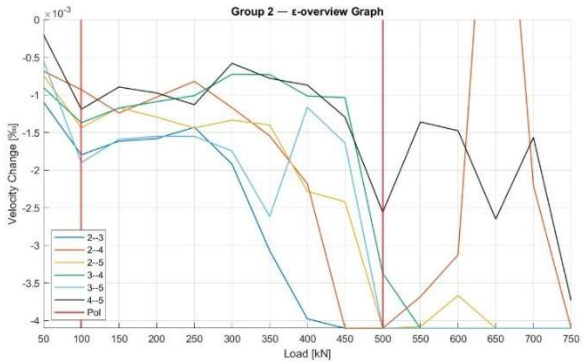


Figure 49, Beam S10H1D Group 2 ε-overview Graph

The POI at 500 kN coincides with the first appearance of the highlighted flexural crack left of SA3. At this load level, another crack can also be observed to the right of SA3. At this load, the SA pairs showing the largest ε reductions do not fully correspond to the location of the observed cracks.

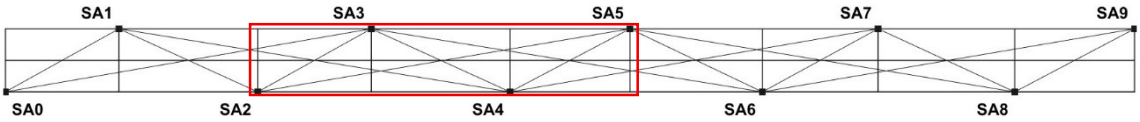
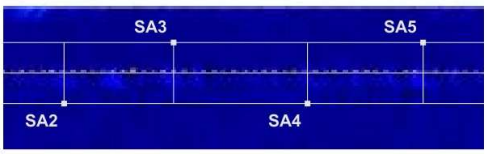
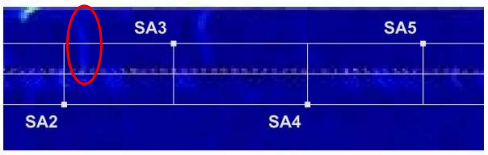


Figure 50, Beam S10H1D, (a) Group location on the beam, (b)-(c) DIC observations



(b)- 100[kN]



(c)- 500[kN]

For Group 3, ε-based POI occur at 250 kN, 500 kN and 650 kN. At 250 kN and 500 kN, ε-values show reductions across the monitored region without visible cracking. At 650 kN, ε again shows a pronounced response and the reference observation shows a flexural crack located near SA4.

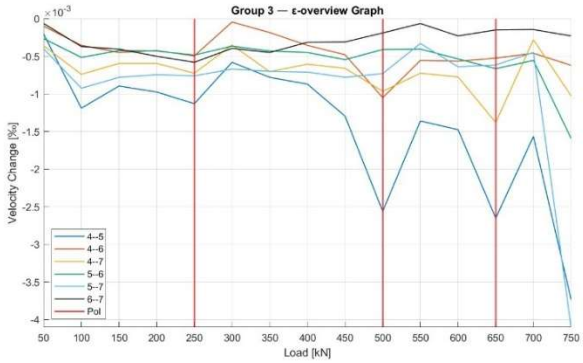


Figure 51, Beam S10H1D Group 3 ε-overview Graph

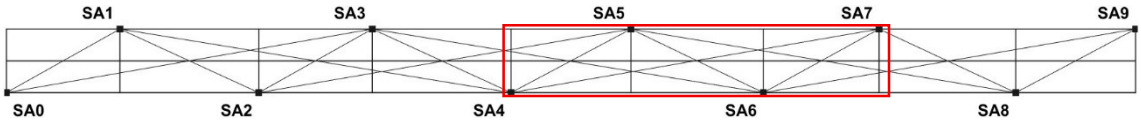
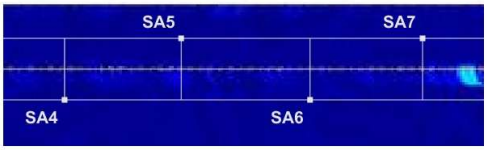
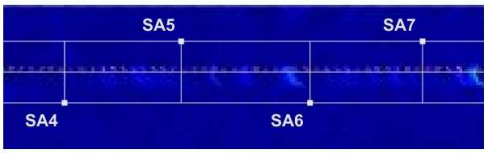


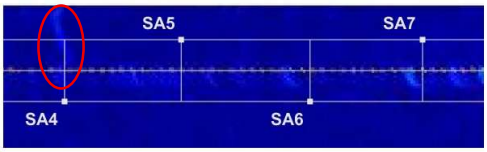
Figure 52, Beam S10H1D, (a) Group location on the beam, (b)-(d) DIC observations



(b)- 250[kN]



(c)- 500[kN]



(d)- 650[kN]

For Group 1, CC-based POI occur at 450 kN, 550 kN, 650 kN and 700 kN. At 450 kN, CC-values exhibit a clear reduction across several SA pairs. The reference observation shows the appearance of multiple flexural cracks, including cracks located near SA0 and SA1, as well as an additional crack outside the monitored area near SA3. At 550 kN, CC-values again show a reduction across the monitored SA pairs, the DIC shows the formation of a new shear crack near SA0. At 650 kN, CC-values again exhibit a reduction, while the reference observation indicates the appearance of both a new shear crack and an additional flexural crack within the monitored region. At 700 kN, CC-values again show a reduction across several SA pairs; the reference observations indicate continued crack widening at higher load levels.

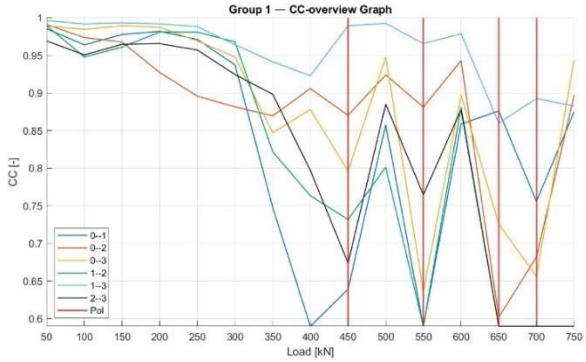


Figure 53, Beam S10H1D Group 1 CC-overview Graph

The POI at 450 kN coincides with the first appearance of multiple flexural cracks within the monitored region. Since several cracks appear across the monitored region at 450 kN, reductions would be expected across most SA pairs; however, this is not consistently reflected in the CC responses as SA pairs 0—2 and 1—3 are above 0.85.
After 450 kN, additional cracking occurs, including the formation of a shear crack at 550 kN and further cracking at 650 kN and 700 kN. While the CC results show clear valleys at these load levels, indicating structural disturbances, the individual SA pairs do not allow clear localisation of specific cracks.

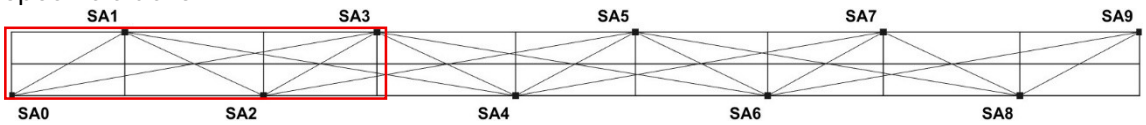
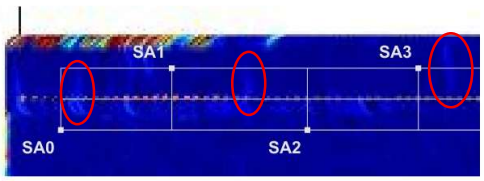
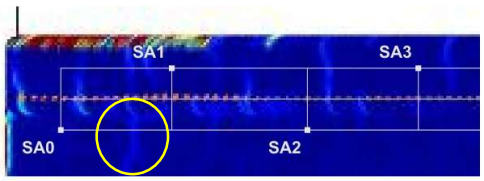


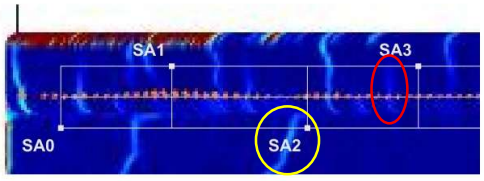
Figure 54, Beam S10H1D, (a) Group location on the beam, (b)-(d) DIC observations



(b)- 450[kN]



(c)- 550[kN]



(d)- 650[kN]

For Group 2, CC-based POI occur at 450 kN, 650 kN and 700 kN. At 450 kN, CC-values exhibit a reduction across multiple SA pairs, with the largest change observed for SA pair 2—3. The corresponding reference observation shows the appearance of a flexural crack near SA3. At 650 kN, CC-values again show a reduction across all SA pairs, most notably for pairs 2—3 and 3—4. The reference observation indicates the formation of a new shear crack near SA2 together with an additional flexural crack near SA3. At 700 kN, CC-values again exhibit a pronounced reduction, no reference observation is available for this load level.

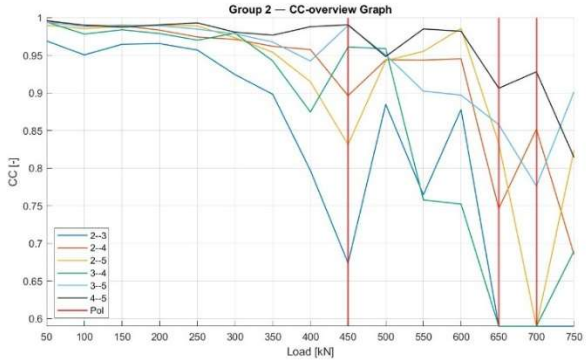


Figure 55, Beam S10H1D Group 2 CC-overview Graph

At 450 kN, SA pair 2—3 shows the strongest reduction and pair 2—5 shows a response close to the threshold; both pairs are located near the crack. However, pair 3—5, which crosses the crack location, does not show a comparable response. At 750 kN multiple additional cracks are observed, localisation through the CC valley is not possible as cracks appear across the entire monitored region.

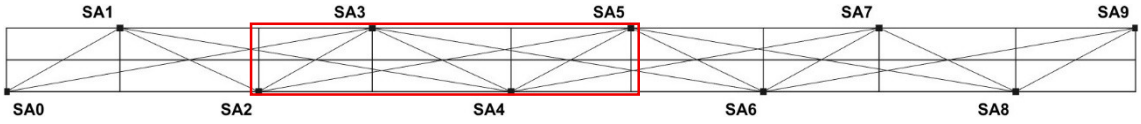
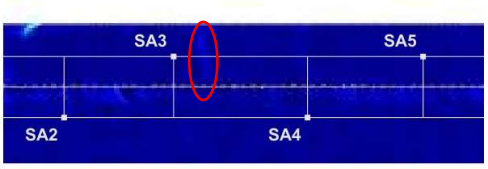
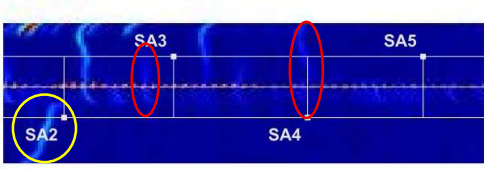


Figure 56, Beam S10H1D, (a) Group location on the beam, (b)-(c) DIC observations



(b)- 450[kN]



(c)- 650[kN]

For Group 3, CC-based POI occur at 500 kN and 650 kN. At 500 kN, CC-values exhibit a reduction across the monitored SA pairs, while no visible cracking is observed in the corresponding reference observation. At 650 kN, CC-values again show a pronounced reduction across the monitored region, with the largest change observed for SA pair 4—5. The reference observation at this load level shows the appearance of a new flexural crack near SA4, while no additional cracks are observed within the monitored region.

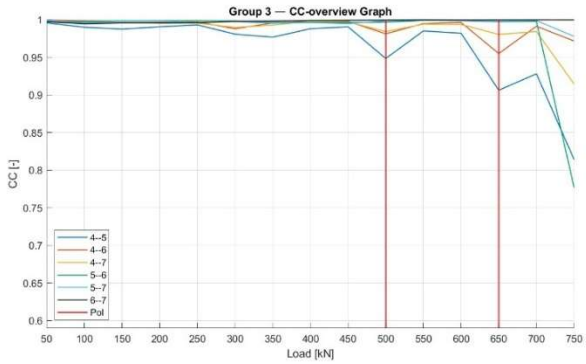


Figure 57, Beam S10H1D Group 3 CC-overview Graph

CC-values remain well above the threshold for most POIs, which may explain why no cracking is observed at those load levels. At 650 kN, SA pair 4—5 shows the lowest value; although it does not fall below the threshold, it coincides with the appearance of the flexural crack near SA4 and is the closest SA pair to this crack.

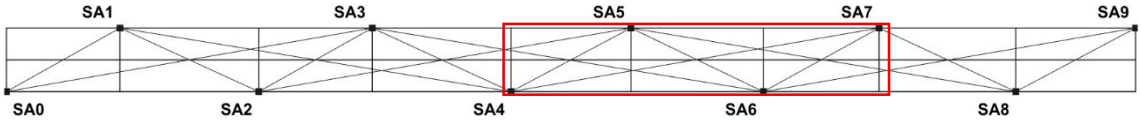
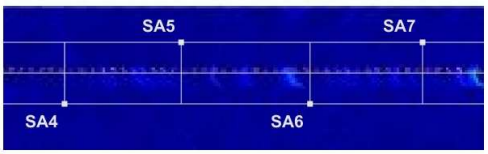
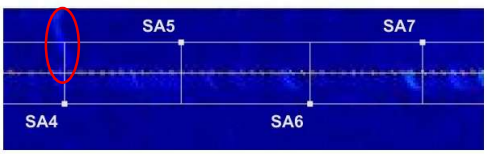


Figure 58, Beam S10H1D, (a) Group location on the beam, (b)-(c) DIC observations



(b)- 500[kN]



(c)- 650[kN]

4.2 – Ray Path Analysis of the Ultrasonic Indicators

In this section, the Ray Path representations are analysed at the individual SA pair level to evaluate ultrasonic indicator behaviour along specific propagation paths. For each beam, the evolution of the ε and CC indicators is examined for the SA pairs within the selected groups. This pair-based assessment enables a more detailed interpretation of ultrasonic responses and their relation to observed crack development.

For beam S10H2D, the analysis is conducted for Groups 1 and 2. For beam S10H2A, Groups 1 and 2 are also examined. Similarly, for beam S10H1D, the Ray Path analysis is performed for Groups 1 and 2. Indicator evolution is interpreted together with the corresponding DIC observations where available.

4.2.1 – Beam S10H2D

For the ray-path analysis of beam S10H2D, Group 1, considering SA pair 1—3, the graph presented below identifies POI at load levels 350 kN, 450 kN, 500 kN and 550 kN, based on observed variations in ε, CC and relative velocity (RV). At 350 kN, both ε and CC exhibit a distinct valley; previous examination most likely indicated cracking within the structural region, although not located along this RP. At 450 kN, this load level marks the initial stage at which RV begins to decrease, which may be associated with the flexural crack observed in the reference observation. At 500 kN, RV decreases below 0.80; no corresponding reference observation is available for this load level. At 550 kN, a pronounced valley in RV is observed, which may be associated with the presence of two flexural cracks within the structural region. Beyond 550 kN, no new cracks are observed; existing cracks widen, while RV does not decrease further. At 650 kN, a possible flexural crack might be the cause of this secondary valley, but the crack does not cross the RP.

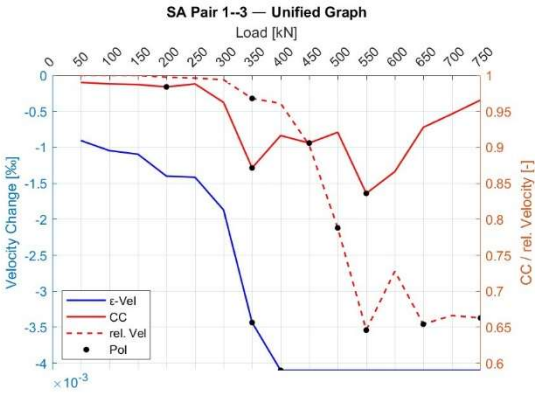


Figure 59, unified graph of Smart Aggregate pair 1—3, Group 1 of Beam S10H2D

The reduction in RV and the development of the flexural cracks match up well. However, there is no clear way to observe the formation of the second flexural crack along the RP through the RV indicator.

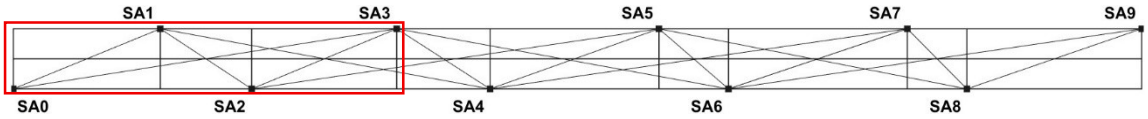
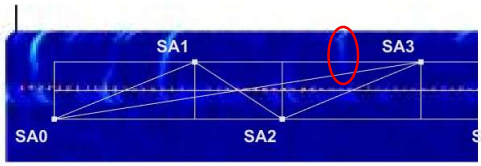
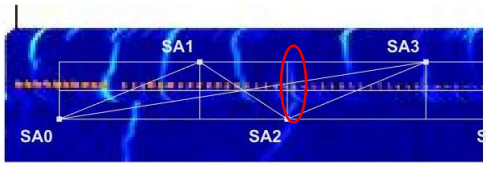


Figure 60, Beam S10H2D, (a) Group location on the beam, (b)-(d) DIC observations



(b)- 450[kN]

(c)- 550[kN]



(d)- 650[kN]

Considering SA pair 0—2, the graph indicates POI at load levels 350 kN, 500 kN, 550 kN, 600 kN and 650 kN. At 350 kN, ε and CC exhibit a valley; no cracking was located along this RP. At 500 kN, ε and CC again exhibit a pronounced valley; no reference observation is available. At 550 kN, RV decreases to approximately 0.80, and at this load a shear crack can be observed along the RP of the SA pair. At 600 kN, RV decreases well below 0.80, while the observed shear crack within the structural region has widened. For load levels beyond 600 kN, a second shear crack becomes visible at 650 kN, which may explain the sustained reduction in RV.

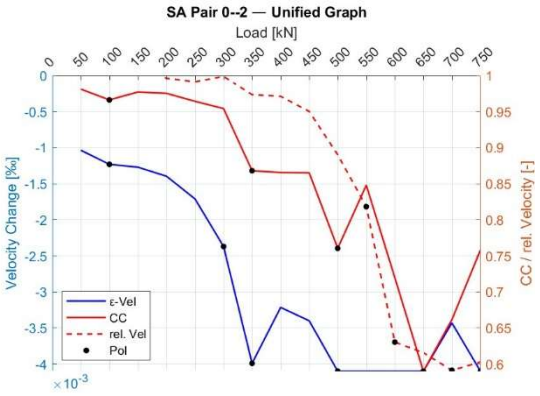


Figure 61, unified graph of Smart Aggregate pair 0—2, Group 1 of Beam S10H2D

The reduction in RV and the development of the shear cracks match up well. Unfortunately, due to the lack of a DIC observation at load 500 kN, the exact initiation of the first shear crack cannot be examined. Again, there is no clear way to observe the formation of the second crack along the RP through the RV indicator.

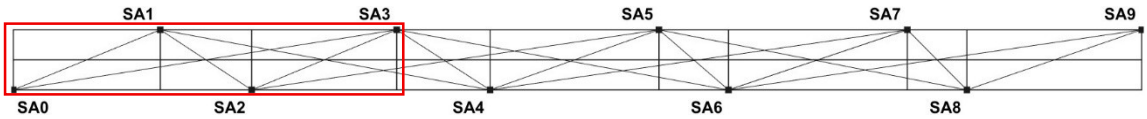
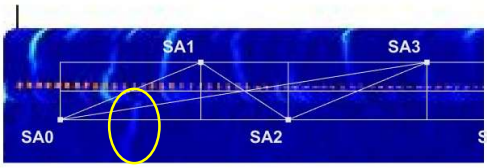
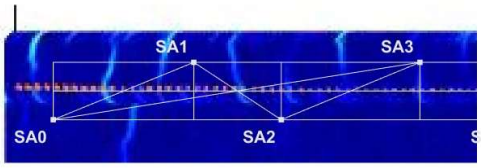


Figure 62, Beam S10H2D, (a) Group location on the beam, (b)-(d) DIC observations



(b)- 550[kN]



(c)- 600[kN]

(d)- 650[kN]

In the graph presented below, SA pair 1—2 identifies POI at load levels 300 kN, 400 kN, 550 kN and 650 kN. At 300 kN, ε decreases below 2.0e-4 while no cracking is observed. At 400 kN, both ε and CC exhibit a valley that may be associated with the more pronounced delamination observed in the reference observation. At 550 kN, CC decreases below 0.80, which may be associated with the first flexural crack observed near the RP. At 650 kN, RV approaches 0.80, which may be associated with the development of a shear crack near the RP; existing cracks become more pronounced at higher load levels.

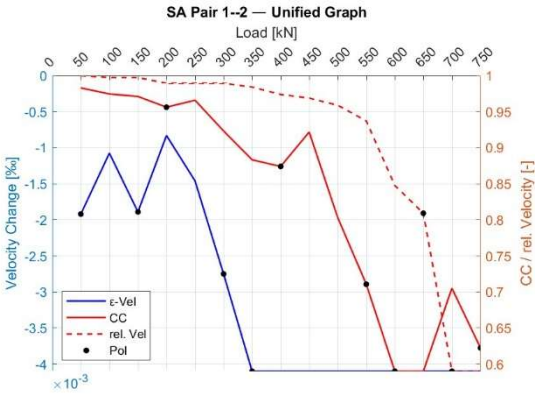


Figure 63, unified graph of Smart Aggregate pair 1—2, Group 1 of Beam S10H2D

By load 650 kN, several cracks appear along RP of SA pair 1—2, while the RV value is still only just above the threshold value of 0.80. It must be noted that none of the cracks cross the RP at a perpendicular angle, and individual cracks cannot be clearly identified through the development of the RV indicator.

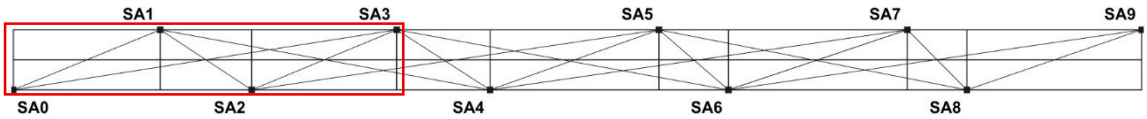


Figure 64, Beam S10H2D, (a) Group location on the beam, (b)-(d) DIC observations

(b)- 400[kN]

(c)- 550[kN]

(d)- 650[kN]

For SA pair 0—3, the graph identifies POI at load levels 350 kN and 450 kN, with continued evolution beyond 600 kN. At 350 kN, ε and CC exhibit a valley, while no cracking was located along this RP. At 450 kN, RV decreases below 0.80, which may be associated with the appearance of two flexural cracks within the structural region. For load levels beyond 600 kN, several additional cracks develop, which may explain the sustained reduction of the ultrasonic indicators.

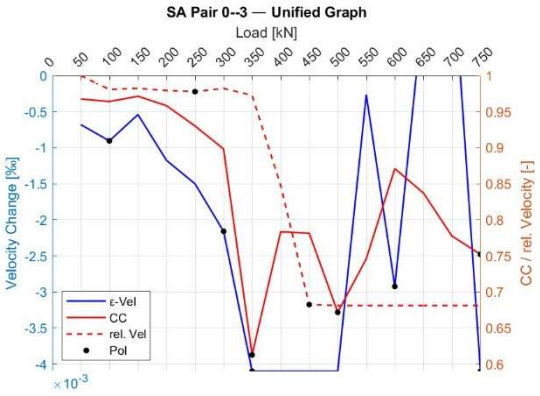


Figure 65, unified graph of Smart Aggregate pair 0—3, Group 1 of Beam S10H2D

By 450 kN, several cracks have developed along the relatively large RP length. However, none cross the RP directly or at a perpendicular angle, yet this still resulted in a RV value below the threshold level.

It must also be noted that this SA pair experienced signal loss, and the arrival times after load 450 kN were manually selected, as described in Chapter 3.3.1.2.

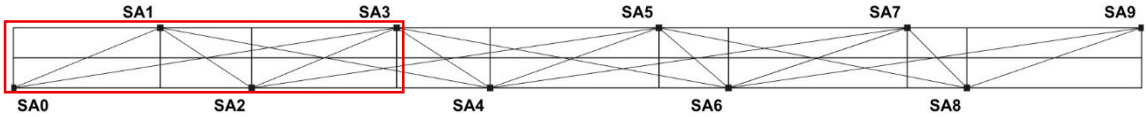
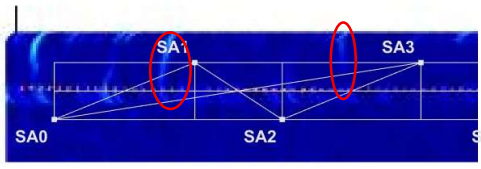


Figure 66, Beam S10H2D, (a) Group location on the beam, (b)-(c) DIC observations



(b)- 450[kN]

(c)- 600[kN]

Considering SA pair 0—1, POI are identified at load levels 350 kN and 550 kN, with continued development beyond 600 kN. At 350 kN, ε and CC exhibit a valley, while no cracking was located along this RP. At 550 kN, RV is near 0.80, which may be associated with multiple cracks developing within the structural region. For higher load levels, no new cracks are observed, although existing cracks become more pronounced.

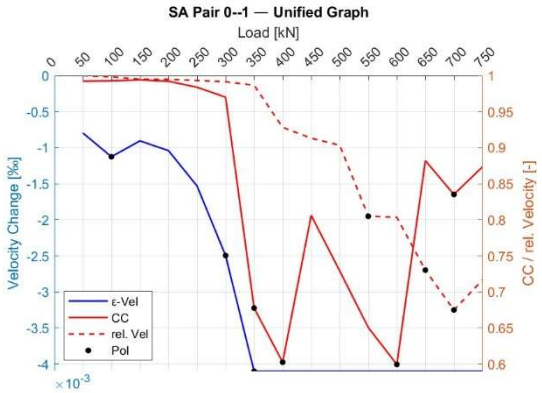


Figure 67, unified graph of Smart Aggregate pair 0—1, Group 1 of Beam S10H2D

Between loads 350 and 550 kN, several cracks develop along the RP. This is reflected in the stepwise reduction of the RV value; however, individual crack identification is again not possible.

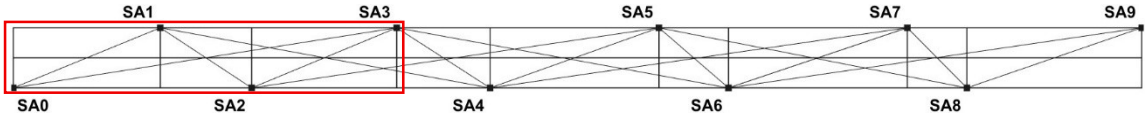
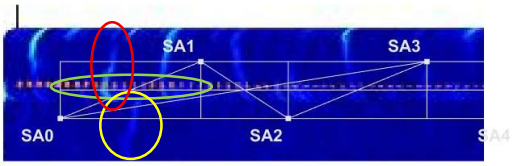


Figure 68, Beam S10H2D, (a) Group location on the beam, (b)-(c) DIC observations

(b)- 350[kN]



(c)- 550[kN]

Finally, for SA pair 2—3, the graph indicates POI at load levels 450 kN, 600 kN and 750 kN. At 450 kN, CC exhibits a valley accompanied by the first major decrease in RV; previous observations indicated delamination (at 400 kN) and a new flexural crack has appeared near the RP, which may explain this response. At 600 kN, CC again exhibits a valley accompanied by a decrease in RV, which may be associated with a flexural crack near the RP. At 750 kN, RV reaches its lowest value, which may be associated with the shear crack observed in the preceding load stage (700 kN) and the overall progression of cracking within the structural region.

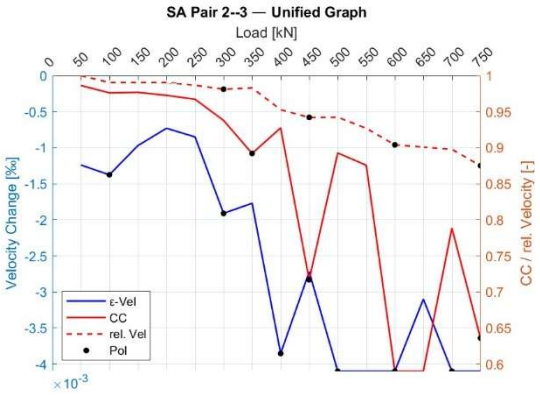


Figure 69, unified graph of Smart Aggregate pair 2—3, Group 1 of Beam S10H2D

Similar to SA pair 0—1, SA pair 2—3 appears to be influenced by several different types of cracks. In this case, the development occurs over a smaller number of cracks but over a longer period, which may explain the steady reduction in the RV value. However, by load 750 kN, the cracks are very pronounced, while the RV value has not yet dropped below the threshold value of 0.80.

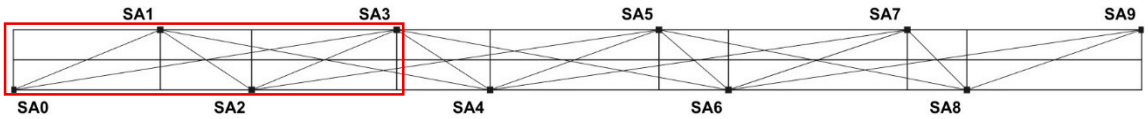
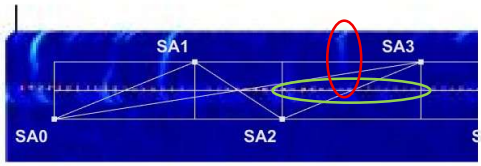
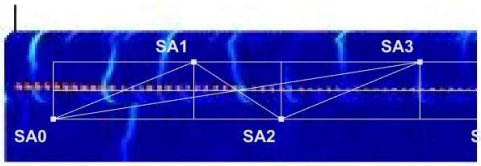


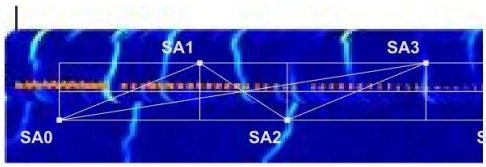
Figure 70, Beam S10H2D, (a) Group location on the beam, (b)-(d) DIC observations



(b)- 450[kN]



(c)- 600[kN]



(d)- 750[kN]

For the ray-path analysis of beam S10H2D, Group 2, considering SA pair 3—5, the graph presented below identifies POI at load levels 450 kN, 600 kN and 650 kN, with continued evolution beyond 700 kN. At 450 kN, both ε and CC exhibit a valley, while no cracking was located along this RP. At 600 kN, CC again exhibits a valley accompanied by the first major decrease in RV, which may be associated with the first flexural crack observed near the RP. At 650 kN, RV approaches 0.80, which may be associated with the flexural crack crossing the RP and the appearance of a second flexural crack. Beyond 700 kN, cracks become more pronounced, consistent with the sustained reduction in RV.

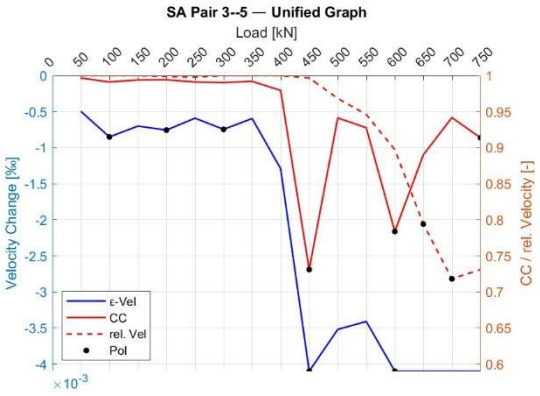


Figure 71, unified graph of Smart Aggregate pair 3—5, Group 2 of Beam S10H2D

Like the top category SA pair 1—3, this top SA pair shows a good correspondence between the reduction in RV and the development of the flexural cracks.

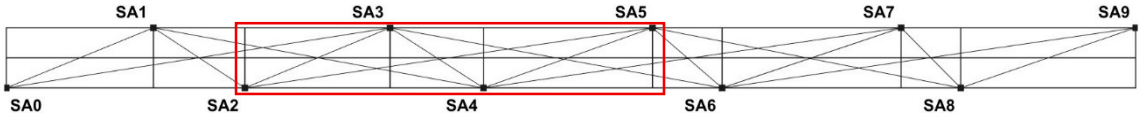
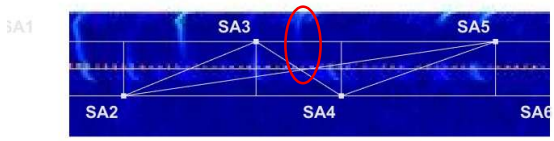
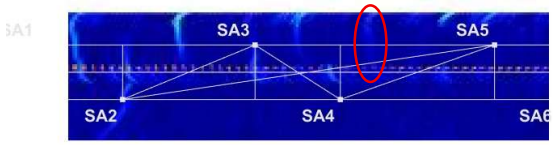


Figure 72, Beam S10H2D, (a) Group location on the beam, (b)-(d) DIC observations

(b)- 450[kN]



(c)- 600[kN]



(d)- 650[kN]

Considering SA pair 2—4, the graph indicates POI at load level 450 kN and from 650 kN onwards. At 450 kN, ε and CC exhibit a valley; identified cracking was not located along this RP. From 650 kN onwards, RV reaches its lowest values, which may be associated with the development of shear cracks near the RP.

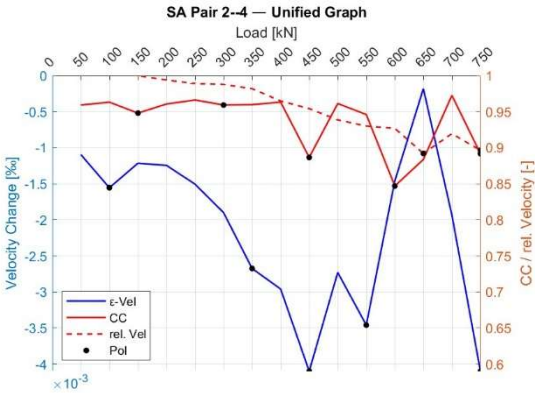


Figure 73, unified graph of Smart Aggregate pair 2—4, Group 2 of Beam S10H2D

The observations show that a shear crack crosses the RP of SA pair 2—4; however, the RV graph does not show a similar reduction as observed in the bottom category SA pair 0—2. This may be due to inaccuracies introduced during the reshaping of the DIC observations.

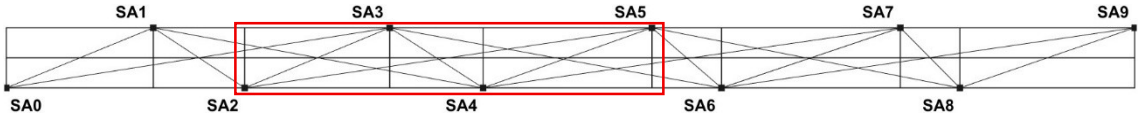
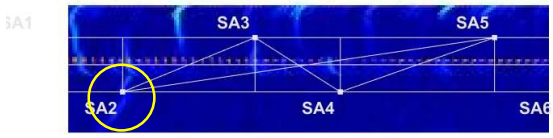


Figure 74, Beam S10H2D, (a) Group location on the beam, (b) DIC observations



(b)- 650[kN]

For SA pair 3—4, POI are identified at load levels 450 kN, 650 kN and 700 kN. At 450 kN, ε and CC exhibit a valley, while observations indicate delamination within the structural region. At 650 kN, CC exhibits a valley which may be associated with cracking near, but not crossing, the RP, while delamination becomes more pronounced. At 700 kN, RV decreases below 0.80, which may be associated with the further widening of the existing cracks.

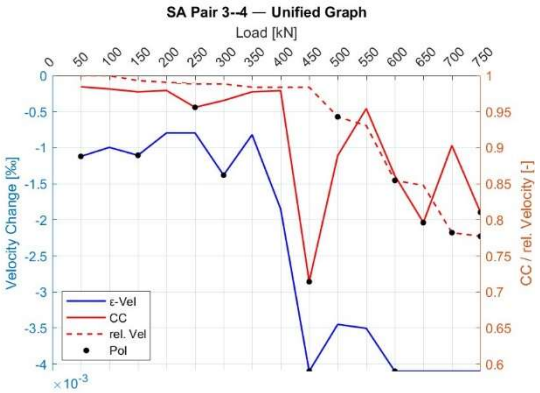


Figure 75, unified graph of Smart Aggregate pair 3—4, Group 2 of Beam S10H2D

The reduction in the RV value does not appear to match the amount and severity of cracking. By load 700 kN, the delamination is quite pronounced, but the flexural crack does not appear to cross the RP of the SA pair.

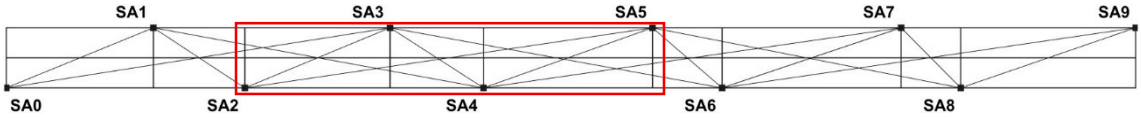
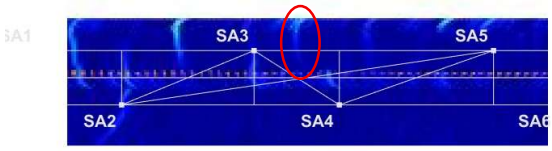
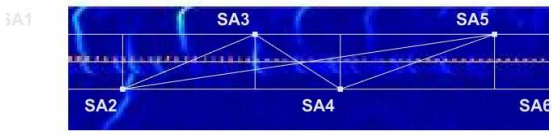


Figure 76, Beam S10H2D, (a) Group location on the beam, (b)-(d) DIC observations

(b)- 450[kN]



(c)- 650[kN]



(d)- 700[kN]

Considering SA pair 2—5, the graph identifies POI at load levels 450 kN, 600 kN and 700 kN. At 450 kN, ε exhibits a valley, while no identified cracking was located along this RP; only possible delamination is observed. At 600 kN, CC decreases below 0.80, accompanied by a major decrease in RV; no new cracks are observed in the immediate region of the RP, only existing cracks have widened. At 700 kN, RV decreases below 0.80, which may be associated with a shear crack near SA2 together with multiple flexural cracks in the surrounding RP region.

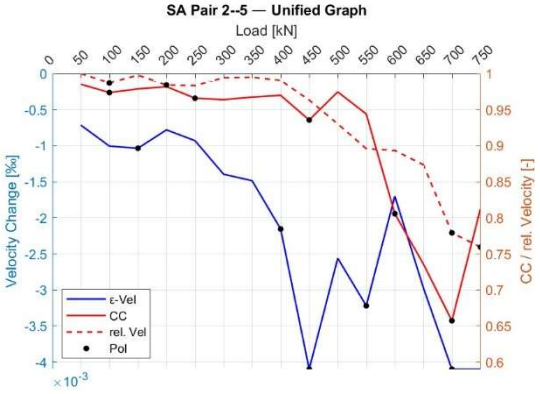


Figure 77, unified graph of Smart Aggregate pair 2—5, Group 2 of Beam S10H2D

Again, for this long diagonal SA pair, the RP intersects several cracks due to its length. The reduction in RV corresponds reasonably well with the development of cracking; however, the individual formation of cracks cannot be identified through the RV indicator.

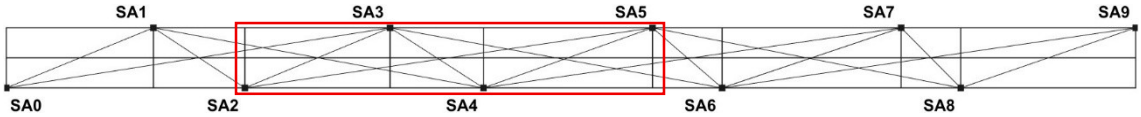
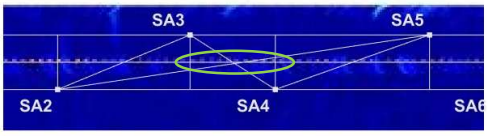
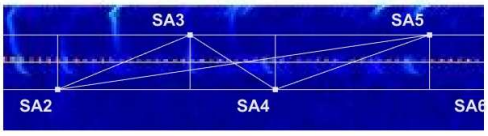


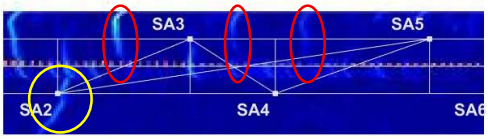
Figure 78, Beam S10H2D, (a) Group location on the beam, (b)-(d) DIC observations



(b)- 450[kN]



(c)- 600[kN]



(d)- 700[kN]

Finally, for SA pair 4—5, POI are identified at load levels 450 kN, 650 kN and 750 kN. At 450 kN, ε and CC exhibit a valley without observed cracking near the RP. At 650 kN, CC reaches its lowest value accompanied by the first decrease in RV, which may be associated with previously observed delamination (600 kN) and the appearance of a flexural crack near the RP. At 750 kN, RV reaches its lowest value, consistent with the progression of cracking within the structural region, although no crack appears to cross the RP other than the delamination.

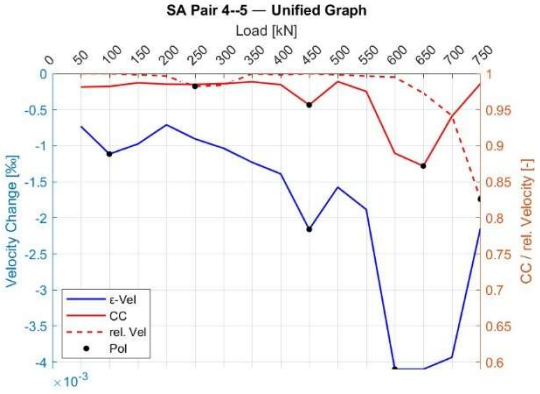


Figure 79, unified graph of Smart Aggregate pair 4—5, Group 2 of Beam S10H2D

The sudden drop in RV cannot be explained by the crack development observed in the DIC measurements. Between loads 650 and 750 kN, no shear or flexural cracks develop that cross the RP of the SA pair. Therefore, the reduction may be solely attributed to the development of delamination, which may not have been properly captured in the DIC observations.

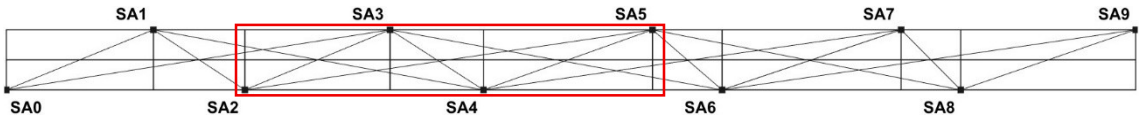
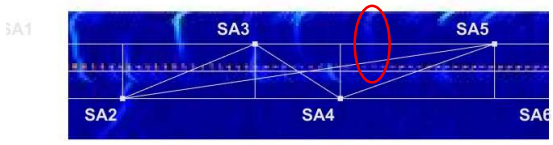
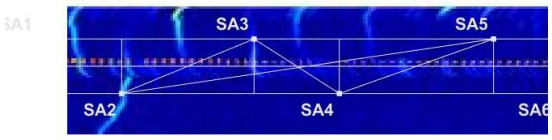


Figure 80, Beam S10H2D, (a) Group location on the beam, (b)-(c) DIC observations



(b)- 650[kN]



(c)- 750[kN]

4.2.2 – Beam S10H2A

For the ray-path analysis of beam S10H2A, Group 1, considering SA pair 1—3, the graph presented below identifies POI at load levels 600 kN, 650 kN and 700 kN, based on observed variations in ε, CC and RV. At 600 kN, ε and CC exhibit a valley; no cracking is identified along this RP. At 650 kN, CC exhibits a valley and RV begins to decrease, which may be associated with the first flexural crack observed within the structural region. At 700 kN, RV decreases below 0.80, which may be associated with the appearance of a second flexural crack and the further widening of the first flexural crack.

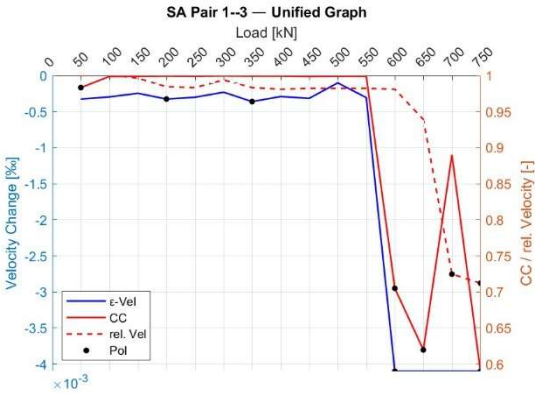


Figure 81, unified graph of Smart Aggregate pair 1—3, Group 1 of Beam S10H2A

Similar to beam S10H2D, the top-category SA pair in this beam shows a good correspondence between the reduction in RV and the development of flexural cracks. However, there is again no clear way to observe the formation of the second flexural crack along the RP through the RV indicator.

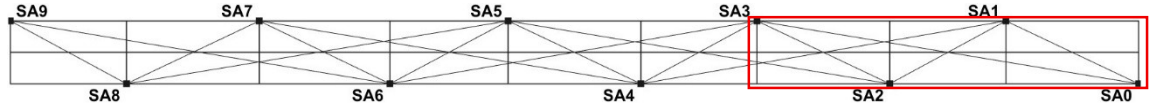
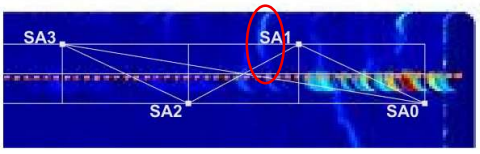
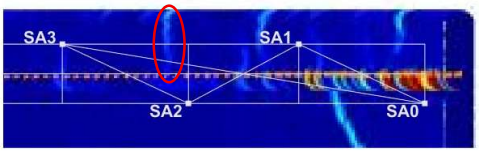


Figure 82, Beam S10H2A, (a) Group location on the beam, (b)-(c) DIC observations



(b)- 650[kN]



(c)- 700[kN]

Considering SA pair 0—2, the graph indicates POI at load levels 500 kN, 600 kN, 650 kN and 700 kN. At 500 kN and 600 kN, ε exhibits a valley without confirmed crack–RP interaction. At 650 kN, CC exhibits a valley, which may be associated with the first shear crack observed in the structural region. At 700 kN, RV decreases below 0.80, which may be associated with the further development of this shear crack.

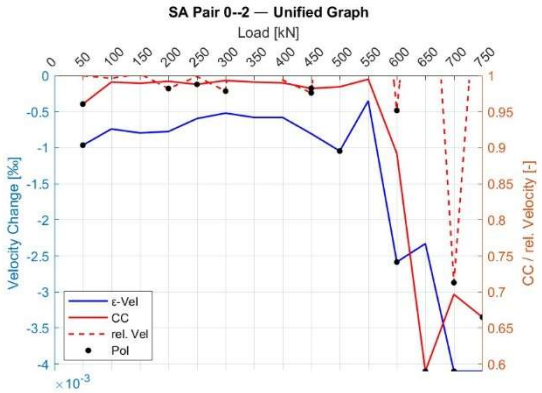


Figure 83, unified graph of Smart Aggregate pair 0—2, Group 1 of Beam S10H2A

It must be noted that the RV results are very irregular; the RV response does not follow a smooth progression but rather fluctuates between load levels. This might have occurred due to signal loss between SA0 and SA2.

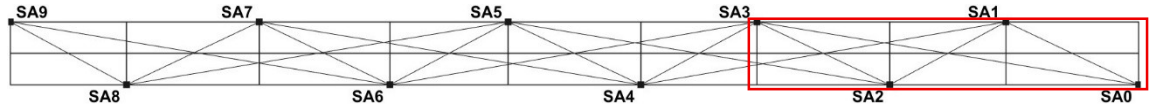
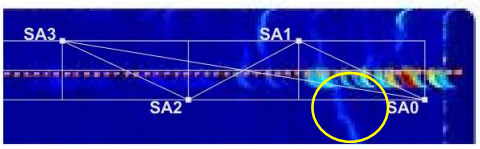
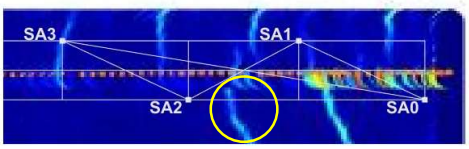


Figure 84, eam S10H2A, (a) Group location on the beam, (b)-(c) DIC observations



(b)- 650[kN]



(c)- 750[kN]

For SA pair 1—2, POI are identified at load levels 600 kN and 700 kN. At 600 kN, ε and CC exhibit a valley without confirmed cracking along the RP, although delamination had been observed at earlier loads (as early as 300 kN). At 700 kN, RV decreases below 0.80, consistent with continued structural degradation; in addition, a new shear crack appears near the RP.

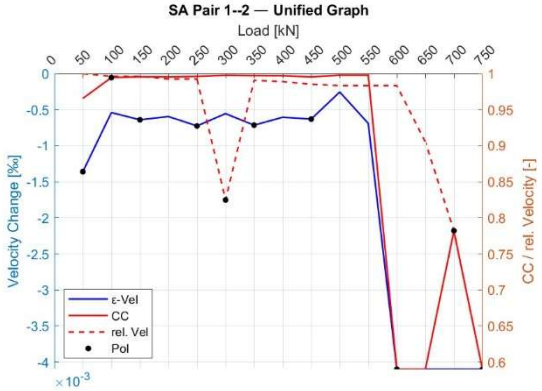
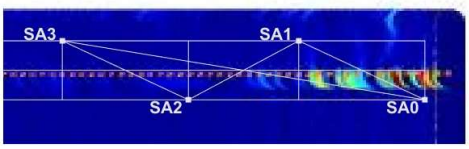


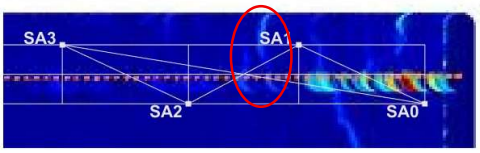
Figure 85, unified graph of Smart Aggregate pair 1—2, Group 1 of Beam S10H2A

The reduction in RV and the development of the flexural and shear cracks match up well. Unfortunately, the onset of the delamination does not appear to influence the RV value.

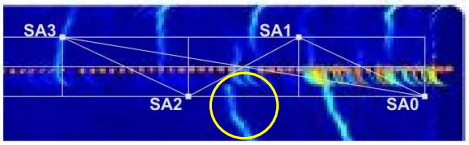
Figure 86, Beam S10H2A, (a) Group location on the beam, (b)-(d) DIC observations



(b)- 600[kN]



(c)- 700[kN]



(d)- 750[kN]

For SA pair 0—3, POI occur at load levels 500 kN, 600 kN, 650 kN and 700 kN. At 500 kN and 600 kN, ε exhibits a valley without confirmed crack–RP interaction. At 650 kN, CC exhibits a valley accompanied by a slight decrease in RV, which may be associated with shear and flexural cracks observed along the RP; delamination had again been observed at an earlier load (300 kN). At 700 kN, RV decreases rapidly, which may be associated with increased crack severity and the development of an additional flexural crack near SA2.

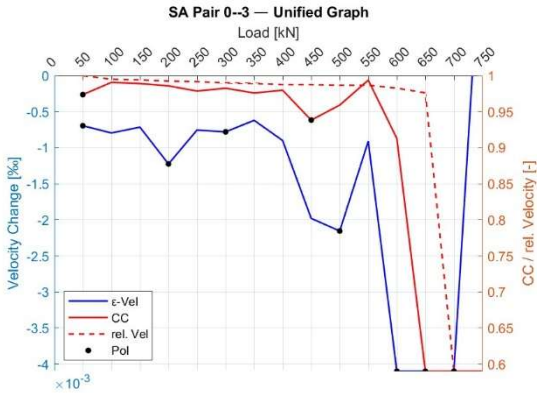
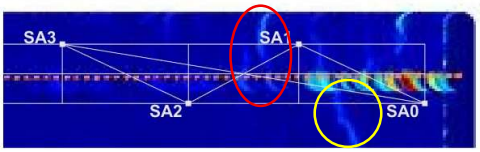


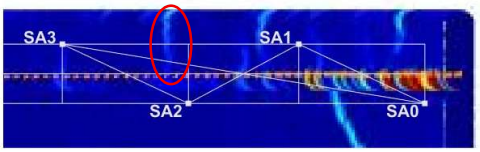
Figure 87, unified graph of Smart Aggregate pair 0—3, Group 1 of Beam S10H2A

At load 650 kN, several cracks have already crossed the RP of the SA pair. However, the RV value only decreases after load 650 kN. Again, the development of interface delamination is not reflected in the RV value.

Figure 88, Beam S10H2A, (a) Group location on the beam, (b)-(c) DIC observations



(b)- 650[kN]



(c)- 700[kN]

Considering SA pair 0—1, POI are identified at load levels 500 kN, 600 kN and 700 kN. At 500 kN, no cracks are observed along the RP; however, delamination was observed at 350 kN. At 600 kN, ε and CC exhibit valleys which may be associated with cracking near, but not crossing, the RP. At 700 kN, RV decreases below 0.80, which may be associated with a shear crack near the RP and the initiation of a flexural crack.

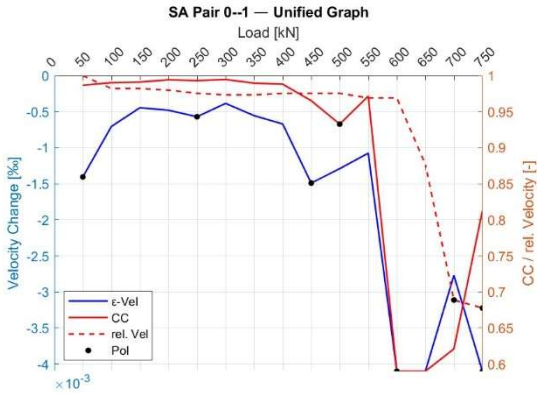


Figure 89, unified graph of Smart Aggregate pair 0—1, Group 1 of Beam S10H2A

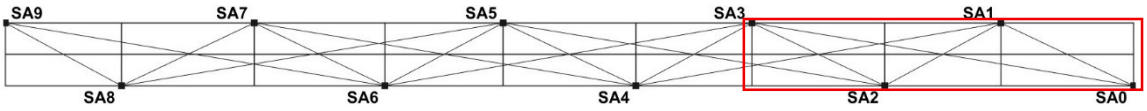
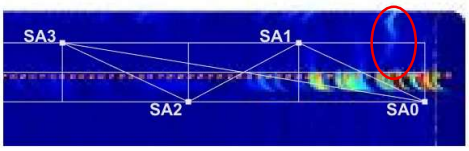
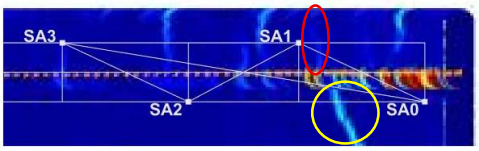


Figure 90, Beam S10H2A, (a) Group location on the beam, (b)-(c) DIC observations



(b)- 600[kN]



(c)- 700[kN]

Finally, for SA pair 2—3, POI occur at load levels 600 kN and 750 kN. At 600 kN, ε exhibits a valley without confirmed cracking near the RP, while delamination had already been observed at earlier load levels no other cracks are observed. At 750 kN, RV remains level near 1.00, however two flexural cracks can be observed along the RP, one of which crosses the RP.

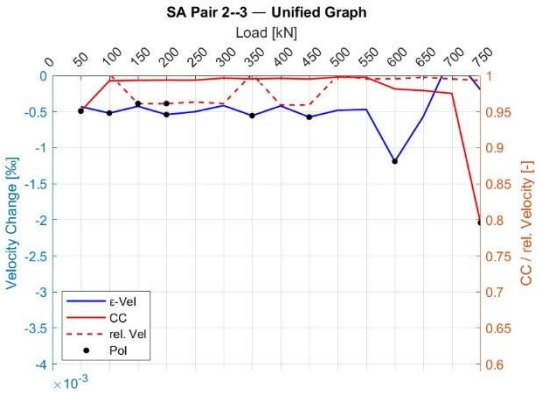


Figure 91, unified graph of Smart Aggregate pair 2—3, Group 1 of Beam S10H2A

Similar to SA pair 0—2, this SA pair also shows signs of signal loss, and both SA pairs share SA2. This might account for the lack of decrease in the RV when the flexural cracks appear.

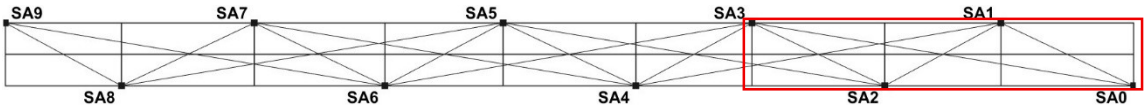
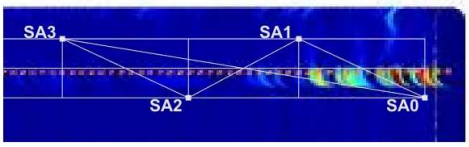
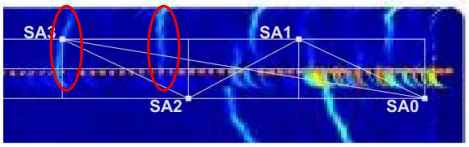


Figure 92, Beam S10H2A, (a) Group location on the beam, (b)-(c) DIC observations



(b)- 600[kN]



(c)- 750[kN]

4.2.3 – Beam S10H1D

For the ray-path analysis of beam S10H1D, Group 1, considering SA pair 1—3, the graph presented below identifies POI at load levels 350 kN, 400 kN and 450 kN, with continued evolution beyond 500 kN. At 350 kN, ε exhibits a valley without visible cracking. At 400 kN, CC exhibits a valley, while cracking is observed outside the RP. At 450 kN, RV decreases below 0.80 without observed cracks crossing the RP. From 500 kN onwards, flexural cracking develops and becomes more pronounced.

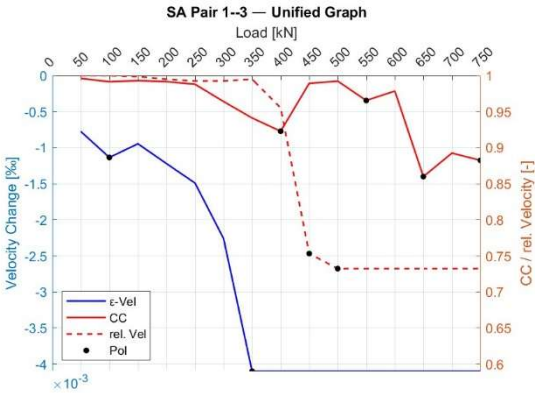


Figure 93, unified graph of Smart Aggregate pair 1—3, Group 1 of Beam S10H1D

This is the first top-category SA pair that does not appear to show a good correspondence between the reduction in RV and the development of the flexural cracks.

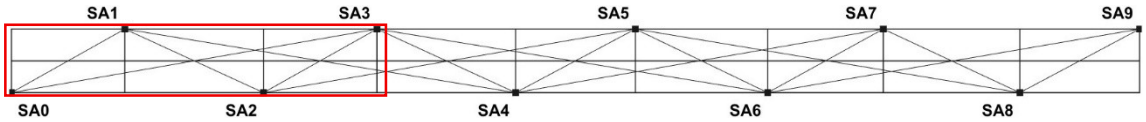
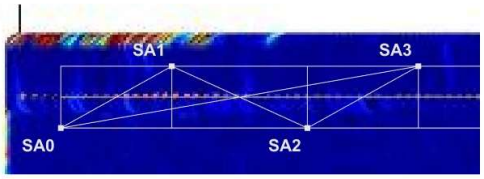


Figure 94, Beam S10H1D, (a) Group location on the beam, (b)-(c) DIC observations



(b)- 450[kN]

(c)- 500[kN]

Considering SA pair 0—2, POI are identified at load levels 350 kN, 500 kN, 550 kN and 650 kN. At 350 kN, ε and CC exhibit valleys without visible cracking. At 500 kN, RV decreases below 0.80 without observed cracks crossing the RP. At 550 kN, ε and CC exhibit valleys accompanied by a further decrease in RV; at this load level the first shear crack is observed along the RP. At 650 kN, RV reaches its lowest value, which may be associated with the appearance of a second shear crack.

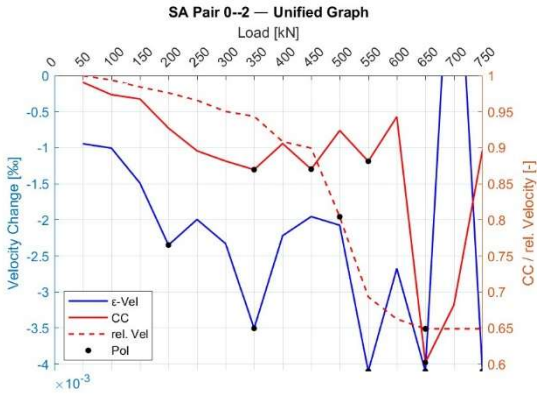


Figure 95, unified graph of Smart Aggregate pair 0—2, Group 1 of Beam S10H1D

Like the other two beams, the bottom-category SA pair shows a good correspondence between the RV progression and the development of cracking from load 550 kN onward. However, it must be noted that at 550 kN the RV value is already relatively low, while no visible shear crack is observed at that stage.

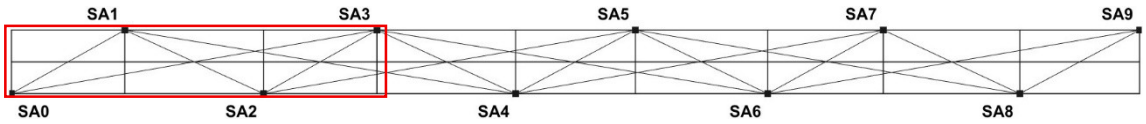
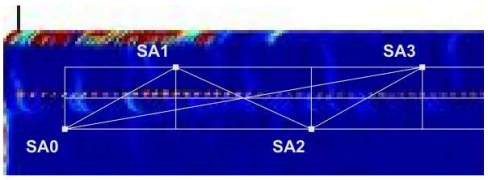
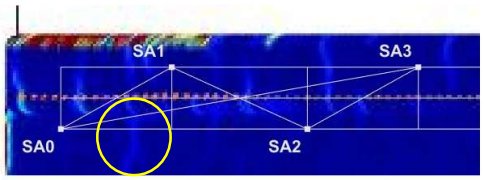


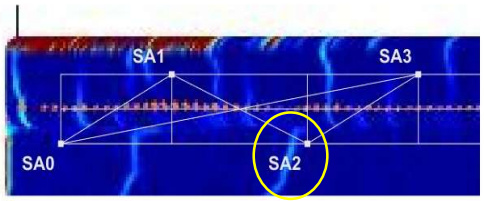
Figure 96, Beam S10H1D, (a) Group location on the beam, (b)-(d) DIC observations



(b)- 500[kN]



(c)- 550[kN]



(d)- 650[kN]

For SA pair 1—2, POI occur at load levels 300 kN, 450 kN, 550 kN and 650 kN. At 300 kN, ε decreases below -2.0e-4 without visible cracking. At 450 kN and 550 kN, CC exhibits valleys accompanied by decreases in RV, while a flexural crack appears to cross the RP at 450 kN. The DIC observations also highlight delamination as early as 400 kN. At 650 kN, RV approaches 0.80, which may be associated with the appearance of a second flexural crack and the first shear crack along the RP.

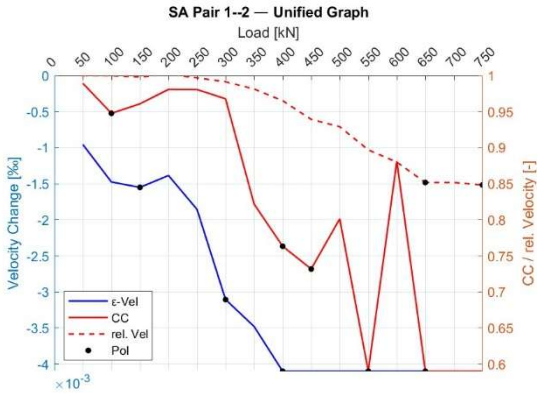


Figure 97, unified graph of Smart Aggregate pair 1—2, Group 1 of Beam S10H1D

Similar to the other beams, this diagonal SA pair is influenced by several different types of cracks, yet the RV value never reaches the threshold value. Additionally, it was again not possible to distinguish the crack types through the RV indicator, and the initiation of the delamination is not properly reflected in the RV value.

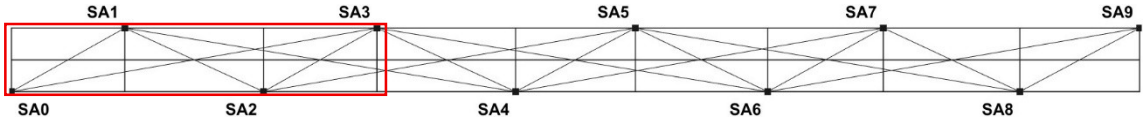
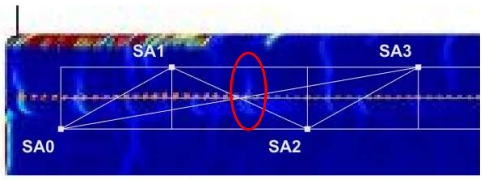
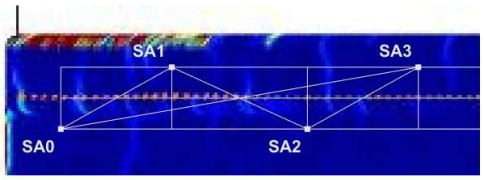


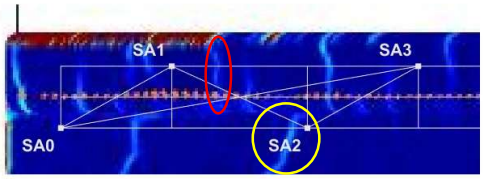
Figure 98, Beam S10H1D, (a) Group location on the beam, (b)-(d) DIC observations



(b)- 450[kN]



(c)- 550[kN]



(d)- 650[kN]

Considering SA pair 0—3, POI are identified at load levels 450 kN, 500 kN and 550 kN. At 450 kN, ε and CC exhibit valleys without observed cracks crossing the RP; however, delamination can be observed at this load level. At 500 kN, RV decreases below 0.80, which may be associated with flexural cracks, including one crossing the RP near SA2 and another near SA0. At 550 kN, CC exhibits a valley accompanied by a further decrease in RV, which may be associated with shear and flexural cracks appearing and crossing the RP.

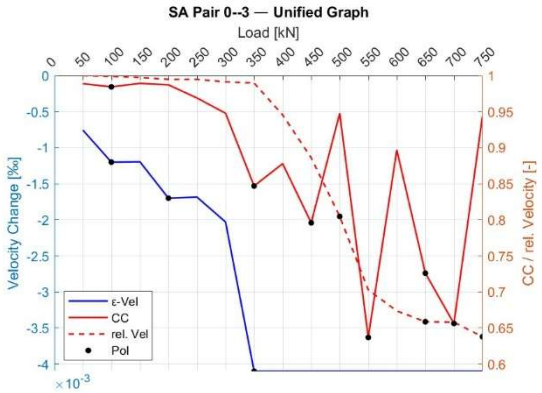


Figure 99, unified graph of Smart Aggregate pair 0—3, Group 1 of Beam S10H1D

Unlike the other diagonal SA pairs, SA pair 0—3 appears to reach the threshold value of 0.80 while the cracks along its RP are still relatively limited. Moreover, the RV value appears to react to the initiation of delamination at load 450 kN.

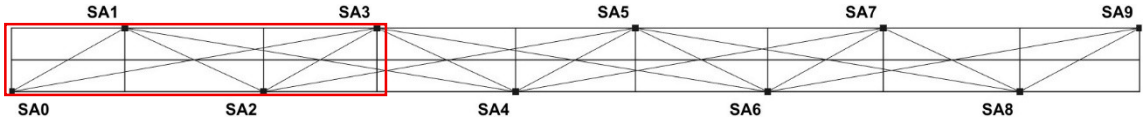
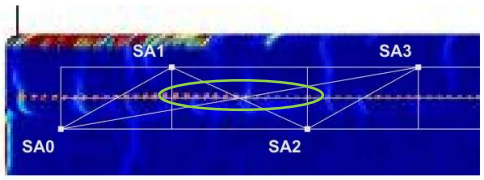
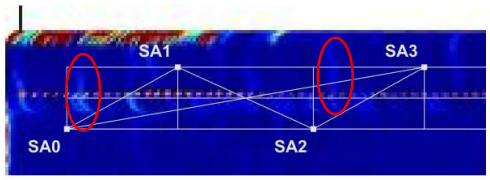


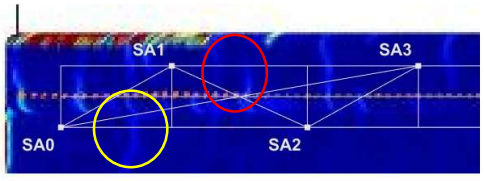
Figure 100, Beam S10H1D, (a) Group location on the beam, (b)-(d) DIC observations



(b)- 450[kN]



(c)- 500[kN]



(d)- 550[kN]

For SA pair 0—1, POI occur at load levels 400 kN, 550 kN and 600 kN. At 400 kN, ε and CC exhibit valleys coinciding with delamination and possible flexural cracking near the RP. At 550 kN, CC exhibits a valley and RV approaches 0.80, which may be associated with shear and flexural cracks crossing the RP. At 600 kN, RV decreases below 0.80, which coincides with multiple cracks crossing the RP.

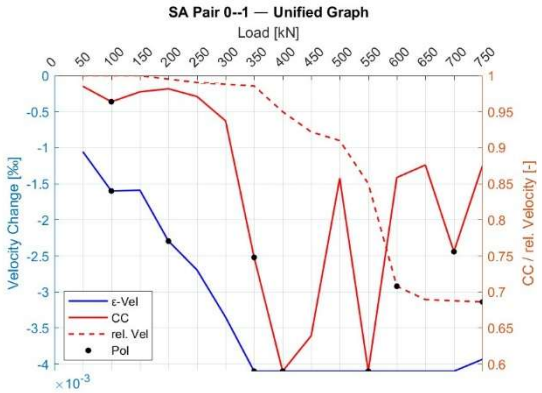


Figure 101, unified graph of Smart Aggregate pair 0—1, Group 1 of Beam S10H1D

Similar to SA pair 0—3, SA pair 0—1 appears to be influenced by the initiation of the delamination process. Subsequently, the threshold value is reached when the flexural and shear cracks observed at load 550 kN cross the RP at load 600 kN.

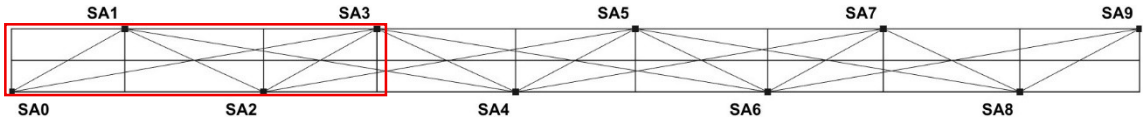
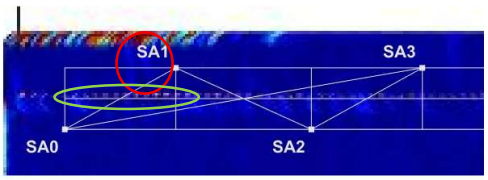
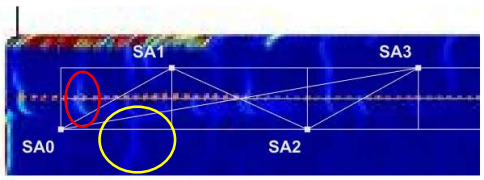


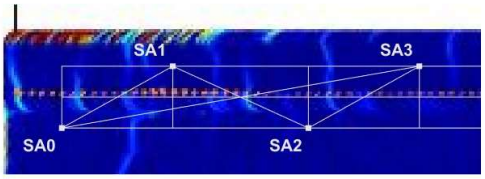
Figure 102, Beam S10H1D, (a) Group location on the beam, (b)-(d) DIC observations



(b)- 400[kN]



(c)- 550[kN]



(d)- 600[kN]

Finally, for SA pair 2—3, POI are identified at load levels 450 kN, 550 kN, 650 kN and 750 kN. At 450 kN, ε and CC exhibit valleys without visible cracking. At 550 kN, CC exhibits a valley associated with delamination and flexural cracking near the RP. At 650 kN, RV approaches 0.90 and a flexural crack crosses the RP. At 750 kN, RV reaches its lowest value, consistent with crack progression.

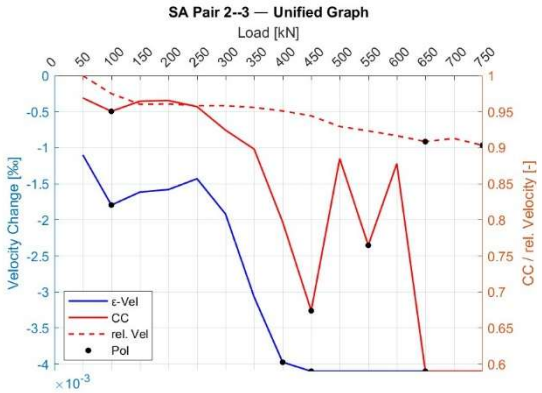


Figure 103, unified graph of Smart Aggregate pair 2—3, Group 1 of Beam S10H1D

This diagonal SA pair behaves more similarly to those observed in the previous two beams than to the other diagonal pairs within this beam.

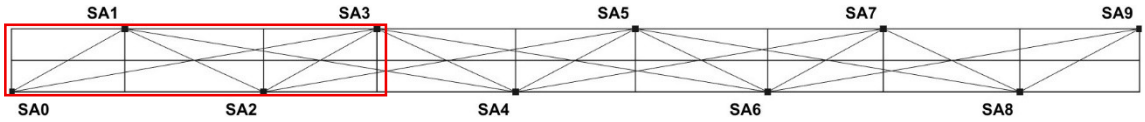
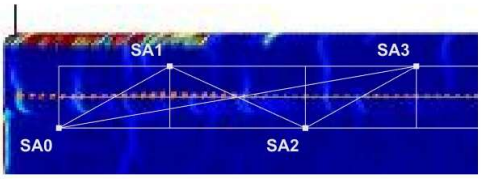
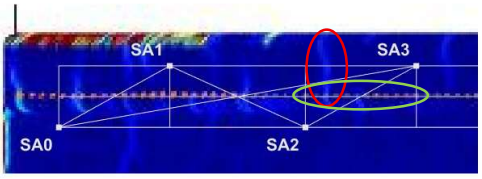


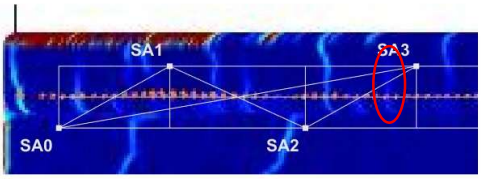
Figure 104, Beam S10H1D, (a) Group location on the beam, (b)-(d) DIC observations



(b)- 450[kN]



(c)- 550[kN]



(d)- 650[kN]

For the ray-path analysis of beam S10H1D, Group 2, considering SA pair 3—5, the graph presented below identifies POI at load levels 350 kN, 400 kN, 500 kN, 650 kN and 750 kN. At early load levels of 350 kN and 400 kN, ε and CC exhibit valleys without visible cracking. At 500 kN, ε exhibits a valley and RV approaches 0.90, which may be associated with the first flexural crack along the RP. At 650 kN and 750 kN, RV decreases below 0.80, which may be associated with additional flexural cracks developing along the RP while existing cracks widen.

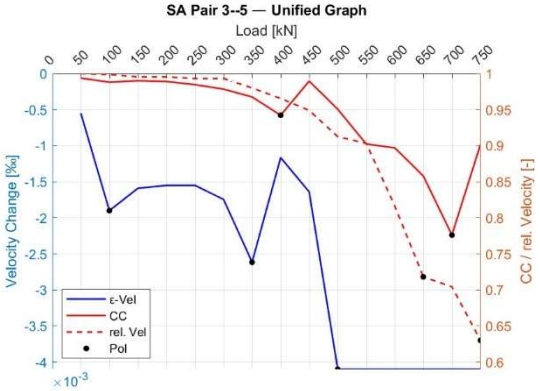


Figure 105, unified graph of Smart Aggregate pair 3—5, Group 2 of Beam S10H1D

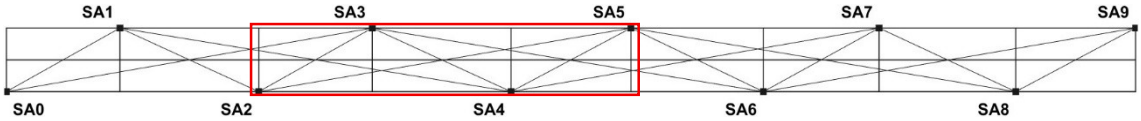
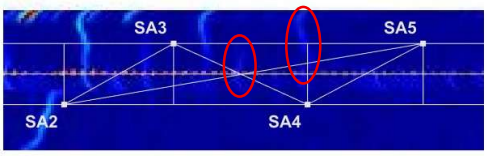
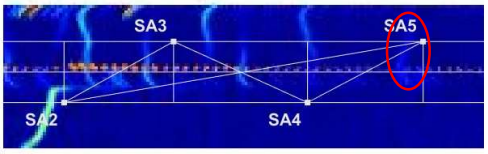


Figure 106, Beam S10H1D, (a) Group location on the beam, (b)-(d) DIC observations

(b)- 500[kN]



(c)- 650[kN]



(d)- 750[kN]

Considering SA pair 2—4, POI occur at load levels 450 kN, 650 kN and 750 kN. Indicator valleys are observed without cracks crossing the RP, while cracking develops near the RP.

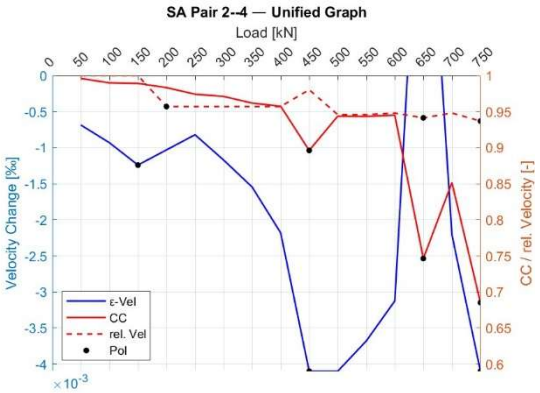


Figure 107, unified graph of Smart Aggregate pair 2—4, Group 2 of Beam S10H1D

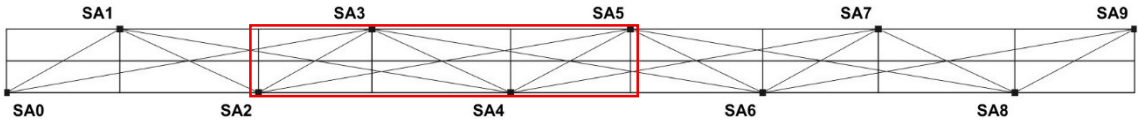
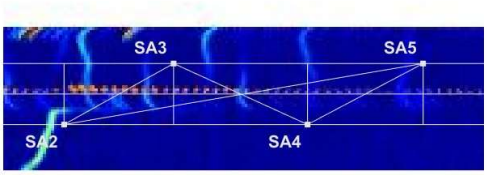


Figure 108, Beam S10H1D, (a) Group location on the beam, (b) DIC observation



(b)- 750[kN]

For SA pair 3—4, POI are identified at load levels 500 kN, 550 kN and 750 kN. At 500 kN, ε exhibits a valley, and at 550 kN, CC exhibits a valley; no cracks appear to cross the RP, although delamination occurs at 500 kN. Early CC valleys occur without visible cracking. Subsequent valleys may be associated with flexural cracking near the RP and progressive delamination. At 750 kN, RV decreases below 0.80, existing cracks widen and a new flexural crack appears.

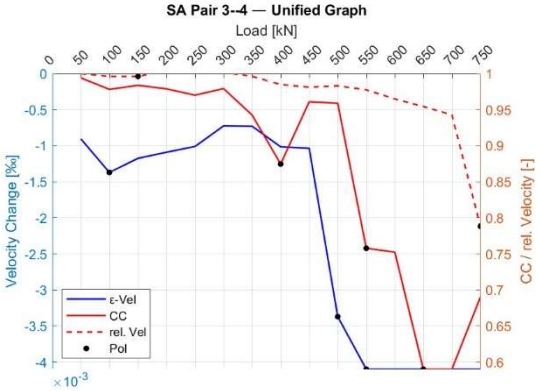


Figure 109, unified graph of Smart Aggregate pair 3—4, Group 2 of Beam S10H1D

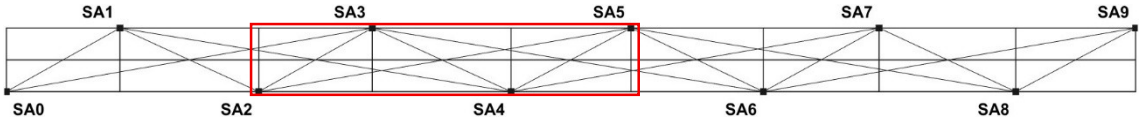
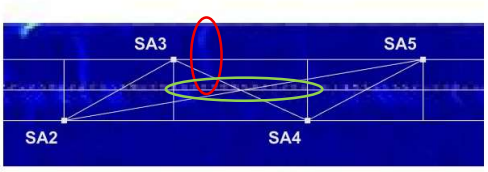
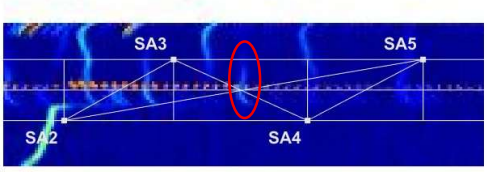


Figure 110, Beam S10H1D, (a) Group location on the beam, (b)-(c) DIC observations



(b)- 500[kN]



(c)- 750[kN]

Considering SA pair 2—5, POI occur at load levels 450 kN, 500 kN and 700 kN. At 450 kN, ε and CC reductions may be associated with flexural cracking crossing the RP. Later indicator behaviour reflects multiple cracks developing near the RP and continued structural degradation; delamination can also be observed at 500 kN. At 750 kN, two additional flexural cracks appear and cross the RP.

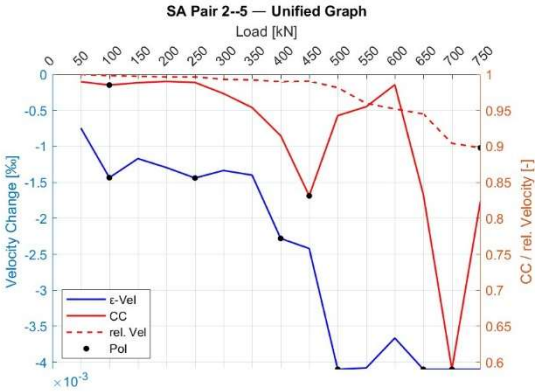


Figure 111, unified graph of Smart Aggregate pair 2—5, Group 2 of Beam S10H1D

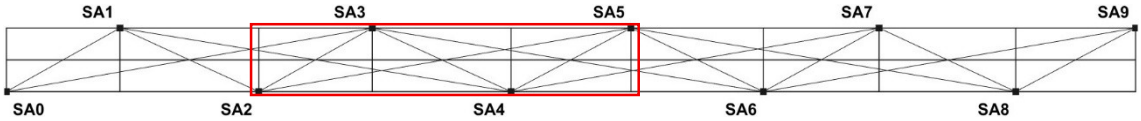
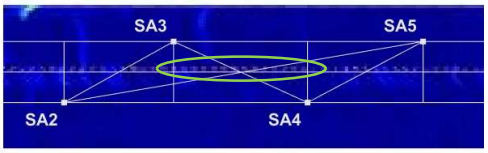
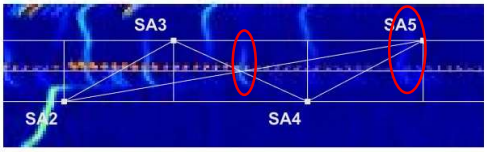


Figure 112, Beam S10H1D, (a) Group location on the beam, (b)-(d) DIC observations

(b)- 450[kN]



(c)- 500[kN]



(d)- 700[kN]

Lastly, for SA pair 4—5, POI are identified at load levels 500 kN, 700 kN and 750 kN. At 500 kN, ε and CC exhibit valleys, with only delamination being observed. At 700 kN, RV exhibits a valley, with no DIC image available. At 750 kN, ε and CC exhibit valleys, with a flexural crack appearing near SA5.

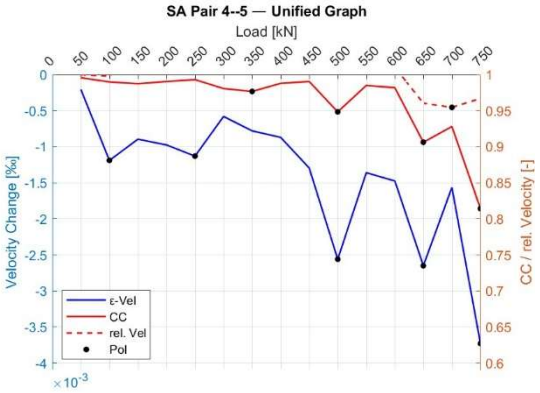


Figure 113, unified graph of Smart Aggregate pair 4—5, Group 2 of Beam S10H1D

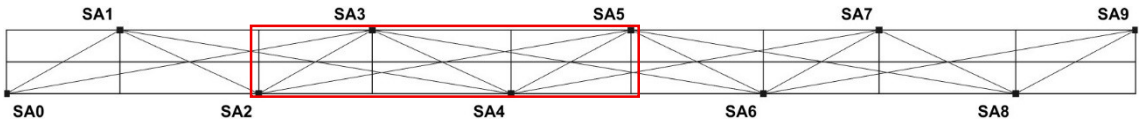
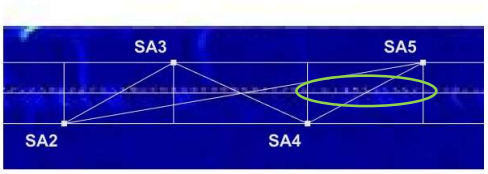
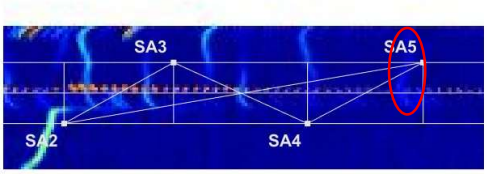


Figure 114, Beam S10H1D, (a) Group location on the beam, (b)-(c) DIC observations



(b)- 500[kN]



(c)- 750[kN]

4.3 – Analysis and Assessment of the Tomographic Reconstruction

This section analyses the Tomographic Reconstruction of the ultrasonic data for the three tested beams. For each beam, the NRMSD value is evaluated to determine up to which load stage the results remain reliable. In addition, the corresponding field velocity and interface time interference results are examined.
For beam S10H2D, tomography results for Groups 1 and 2 and interface results for Groups 1 and 2 are assessed. For beam S10H2A, tomography for Group 1 and interface results for Group 1 are analysed. For beam S10H1D, tomography Groups 1 and 2 and interface Groups 1 and 2 are examined. Where available, the indicator evolution across load levels is interpreted together with the corresponding DIC observations.

4.3.1 – Beam S10H2D

Accuracy Assessment

Figure 115 presents the evolution of the NRMSD values with increasing load. From this graph, two distinct phases can be identified. Between load levels of 0 and 1000 kN, the NRMSD values remain low and relatively stable, indicating that the Tomographic Reconstruction of the ultrasonic data is reliable and the reconstructed fields can be confidently interpreted. After 1000 kN, the NRMSD values begin to increase, suggesting a gradual reduction in reconstruction accuracy, indicating a substantial loss of reliability in the tomographic reconstruction. Consequently, the results obtained after this load level are not further examined.

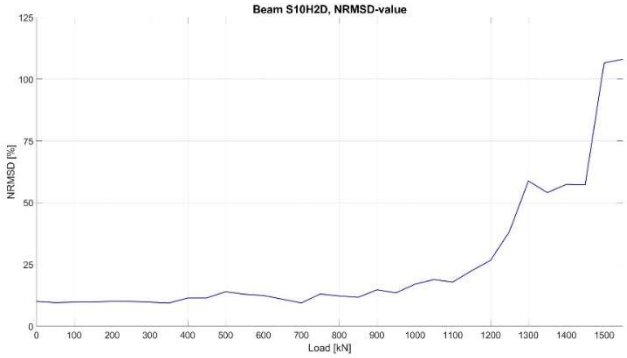


Figure 115, normalised RMSD-value graph of beam S10H2D

Analysis of the Reconstructed Tomographic Indicators

For the tomography analysis of beam S10H2D, Group 1, the Graph presented below identify POI at load levels 450 kN, 650 kN, 700 kN, 850 kN and 900 kN, based on observed variations in relative velocity distribution across the monitored fields. At 450 kN, Field V exhibits a valley; a flexural crack is observed in Field V, while notable cracks are also observed in Fields I and II. At 650 kN, Fields I and II decrease below 0.80, which may be associated with the presence of multiple cracks within these regions.

At 700 kN, Fields III and IV decrease below 0.80; earlier load stages indicated values near 0.80 together with visible cracking, which may explain the observed response. At 850 kN, Fields V and VI decrease below 0.80, consistent with multiple cracks previously observed in these regions and the continued reduction in relative velocity. At 900 kN and higher load levels, all monitored fields decrease below 0.80, while cracks become more pronounced throughout the structural region.

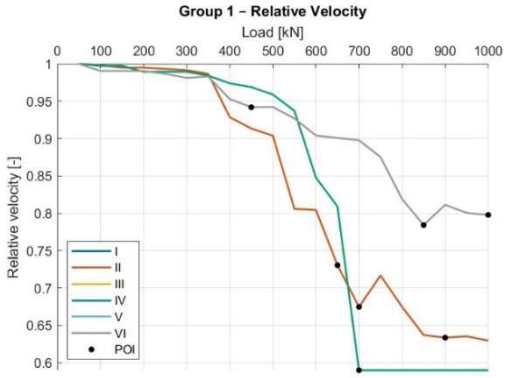


Figure 116, Beam S10H2D Group 1 RV-overview Graph

At 450 kN, no POI are identified in Fields I and II, although the relative velocity has already reduced to approximately 0.90, while cracks are visible in these fields. A similar decreasing trend is observed in Fields III and IV, although no cracks are observed in those regions at this stage. Some fields exhibit reductions in relative velocity before cracks are observed and vice versa, this might be the result of the Tomographic Reconstruction spreading the results over larger areas.
Each vertical pair of fields (i.e., the fields located directly above and below one another) produces identical relative velocity values. This behaviour is consistent across all analysed beams and field configurations. As each field represents a different region, different values would normally be expected. This repeated pairing may therefore indicate a potential issue in the calculation procedure.

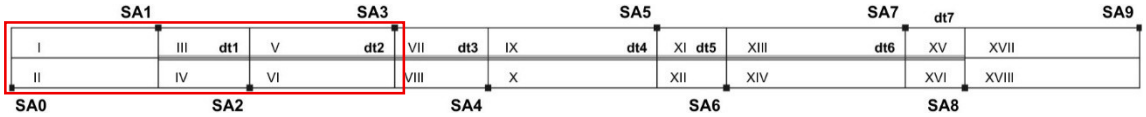
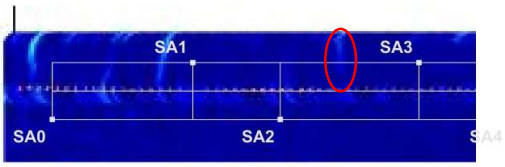
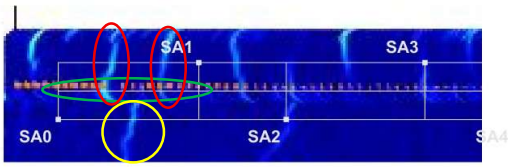


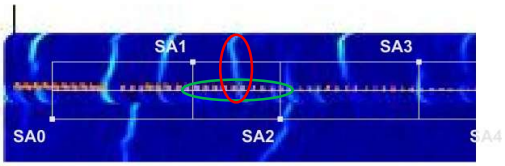
Figure 117, Beam S10H2D, (a) Group location on the beam, (b)-(c) DIC observations



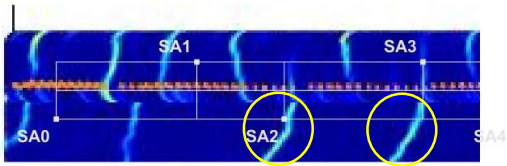
(b)- 450 [kN]



(c)- 650[kN]



(d)- 700[kN]



(e)- 800[kN]

For Group 2, POI are identified at load levels 450 kN, 700 kN and 850 kN. At 450 kN, Fields V and VI exhibit values near 0.95; a flexural crack is observed in Field V. At 700 kN, Fields VII and VIII decrease below 0.80, which may be associated with multiple cracks observed within these regions. At 850 kN, all monitored fields decrease below 0.80, including Fields V and VI, consistent with the presence of multiple cracks previously observed and a continued decrease in relative velocity.

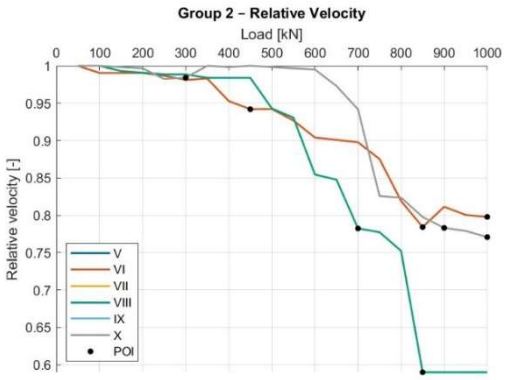


Figure 118, Beam S10H2D Group 2 RV-overview Graph

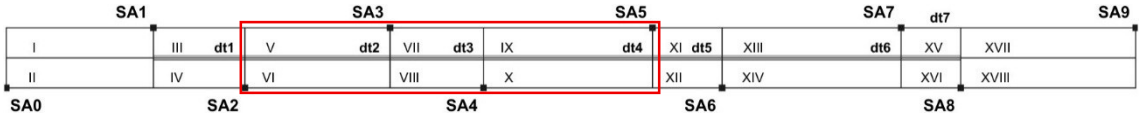
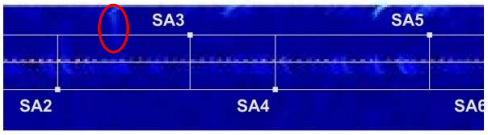
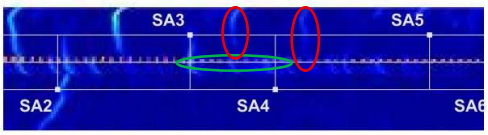


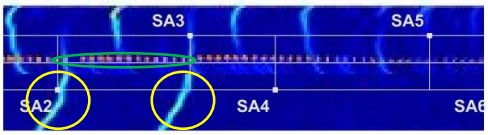
Figure 119, Beam S10H2D, (a) Group location on the beam, (b)-(d) DIC observations



(b)- 450[kN]



(c)- 700[kN]



(d)- 850[kN]

Analysis of the Interface Time Interference

Considering the tomography region of beam S10H2D using dt1–dt3, the graphs indicate POI at 400 kN, 450 kN, 500 kN, 550 kN, 700 kN and 750 kN. At 400 kN, dt1 exceeds 1.5 while dt2 and dt3 remain close to 1.0, with no clear signs of delamination in the reference observations. At 450 kN, dt1 exceeds 3.0 while dt2 and dt3 remain near 1.0; however, the DIC observations do not show clear signs of delamination. At 500 kN, dt1 remains elevated while dt2 increases above 1.5, although no reference observation is available. At 550 kN, dt1 decreases while dt2 exceeds 2.0 and dt3 slightly exceeds 1.0; reference observations indicate clear delamination across dt1 and dt2, while dt3 shows no clear signs.

From 700 kN onwards, dt2 exceeds 2.5 and dt3 exceeds 1.5 while dt1 remains reduced. The continued evolution of these indicators is consistent with progressive delamination observed in the reference observations.

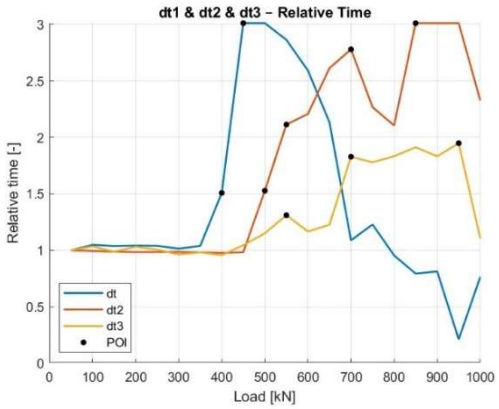


Figure 120, Beam S10H2D dt_1-dt_3 overview Graph

The dt1–dt3 indicators show several early responses before clear delamination becomes visible in the reference observations. At later stages, when delamination becomes more pronounced, dt2 and dt3 exhibit elevated values simultaneously while dt1 decreases. This behaviour may be related to the larger structural deformations at higher load levels, which could affect the tomographic reconstruction.

A dt value of approximately 2.0 appears to indicate the presence of delamination, based on the correspondence between elevated indicator values and the observed interface separation.

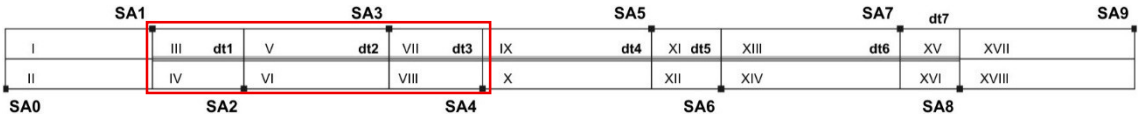
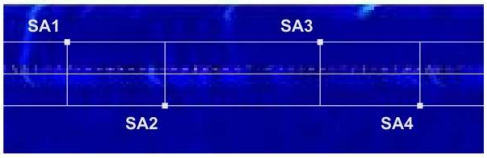
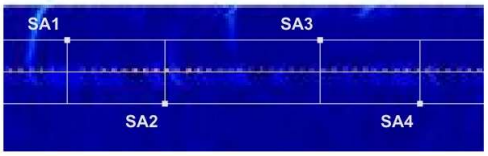


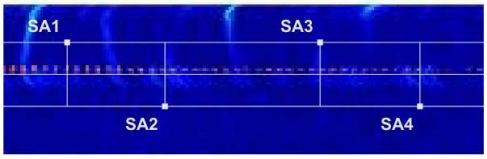
Figure 121, Beam S10H2D, (a) Group location on the beam, (b)-(f) DIC observations



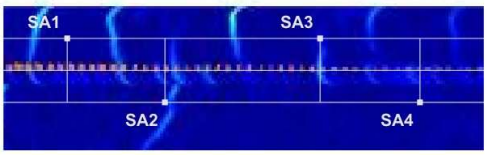
(b)- 400[kN]



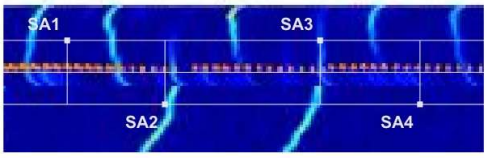
(c)- 450[kN]



(d)- 550[kN]



(e)- 700[kN]



(f)- 850[kN]

For the tomography region using dt3–dt5 in beam S10H2D, POI are identified at 550 kN, 600 kN, 700 kN and 800 kN. At 550 kN, dt3 exceeds 1.0 and dt5 exceeds 2.0, although the reference observations show no clear signs of delamination. At 600 kN, dt4 increases above 2.5 while dt3 and dt5 remain elevated; however, reference observations still indicate no clear delamination. At 700 kN, dt3 exceeds 1.5 while dt4 and dt5 remain elevated. The reference observations show no clear delamination in the dt3 region, while dt4 and dt5 exhibit visible delamination. At 800 kN, dt4 decreases while dt3 and dt5 remain elevated, consistent with sustained internal damage and progressive delamination. At this load level, all dt regions show signs of delamination.

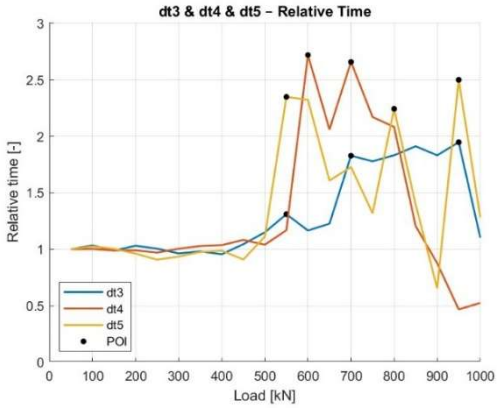


Figure 122, Beam S10H2D dt_3-dt_5 overview Graph

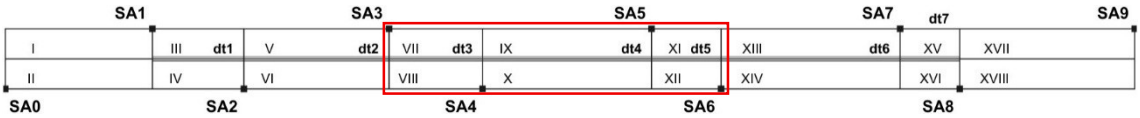
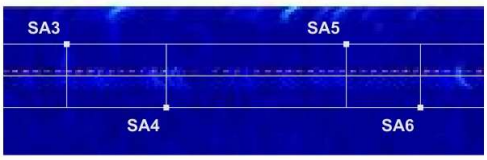
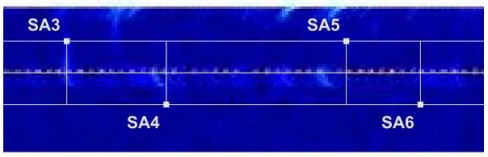


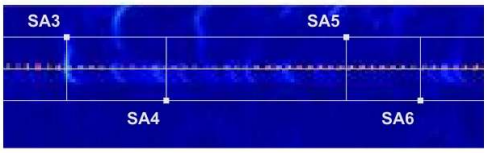
Figure 123, Beam S10H2D, (a) Group location on the beam, (b)-(e) DIC observations



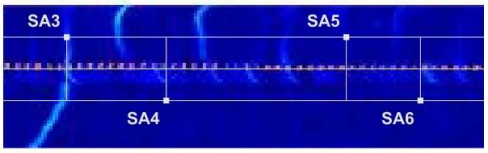
(b)- 550[kN]



(c)- 600[kN]



(d)- 700[kN]



(e)- 800[kN]

4.3.2 – Beam S10H2A

Accuracy Assessment

Figure 124 presents the evolution of the NRMSD values with increasing load. The NRMSD values remain low and relatively stable throughout the whole dataset, indicating that the tomographic reconstruction of the ultrasonic data is reliable and the reconstructed fields can be confidently interpreted.

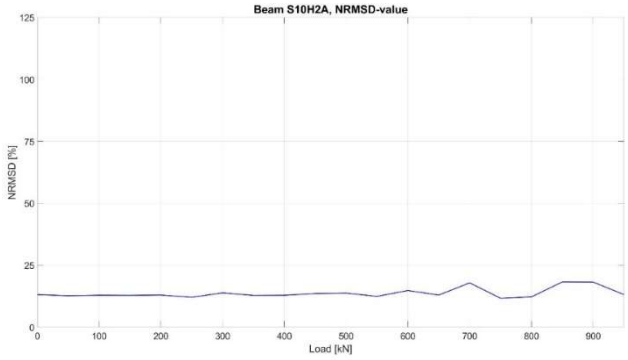


Figure 124, normalised RMSD-value graph of beam S10H2A

Analysis of the Reconstructed Tomographic Indicators

For the tomography analysis of beam S10H2A, Group 1, the Graph presented below identify POI at load levels 650 kN, 700 kN, 750 kN and 800 kN. At 650 kN, Fields I–IV exhibit values near 0.90, while multiple cracks are observed in Fields I and II. At 700 kN, Fields I–IV decrease toward 0.80, with cracks in Fields I and II becoming more pronounced and cracks in Fields III and IV already visible in earlier observations. At 750 kN, a pronounced indicator valley is observed across Fields I–IV, which may be associated with the appearance of a new shear crack in Field V. At 800 kN, Fields IV and V exhibit values near 0.90, consistent with previously observed cracking in Fields V and VI.

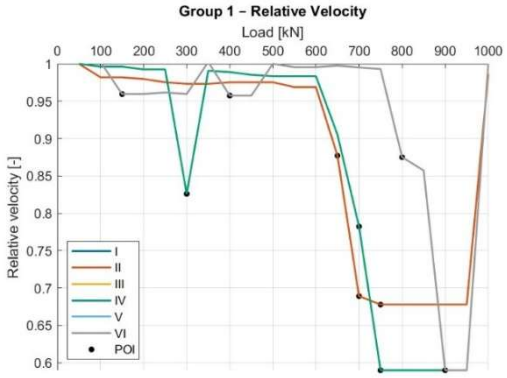


Figure 125, Beam S10H2A Group 1 RV-overview Graph

In this beam, the translated tomographic indicators appear to respond primarily during later stages of the loading process, when cracking has already developed within several fields.

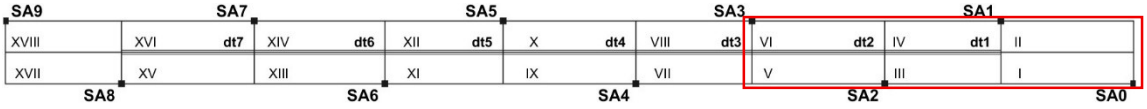
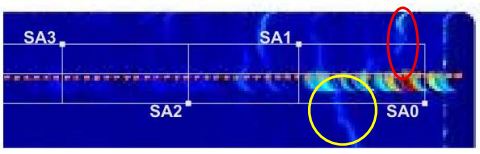
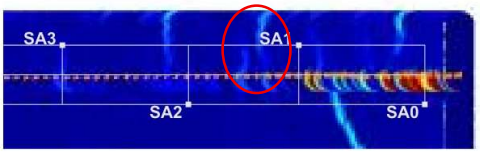


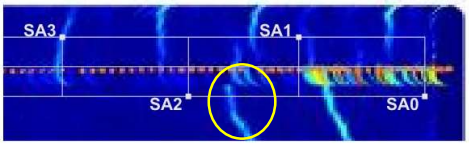
Figure 126, Beam S10H2A, (a) Group location on the beam, (b)-(d) DIC observations



(b)- 650[kN]



(c)- 700[kN]



(d)- 750[kN]

Analysis of the Interface Time Interference

Considering the tomography region using dt1–dt3 for beam S10H2A, POI occur at 300 kN, 650 kN, 700 kN, 750 kN, 800 kN and 900 kN. At 300 kN, all delay-time indicators remain below 1.0, with no clear signs of internal damage. At 650 kN, dt1 remains below 1.0 while dt2 and dt3 remain close to 1.0, although reference observations indicate clear delamination. At 700 kN, dt1 increases to approximately 3.0 while dt2 and dt3 remain near 1.0, which is inconsistent with the observed delamination. At 750 kN and 800 kN, dt1 decreases while dt2 increases and dt3 remains near 1.0; internal delamination remains clearly observable. At 900 kN, all dt indicators decrease while internal damage remains evident in the reference observations.

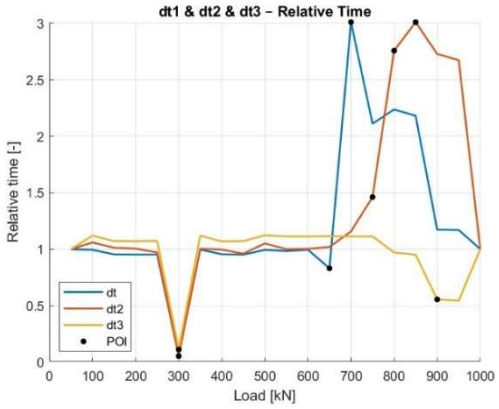


Figure 127, Beam S10H2A dt_1-dt_3 overview Graph

The behaviour of the dt indicators in this beam shows a weaker correspondence with the observed delamination. Instances occur where delamination is clearly visible while the dt indicators remain relatively unchanged.
As observed in the field velocity graph, here all dt indicators show a sudden decrease at 300 kN. This behaviour may indicate a calculation issue in the tomographic reconstruction.

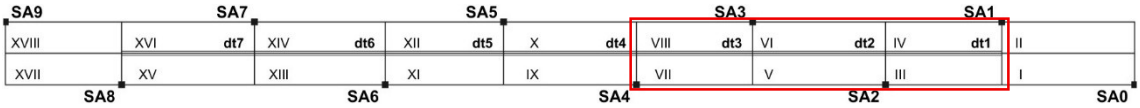
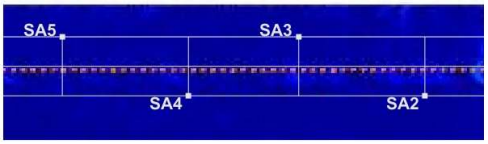
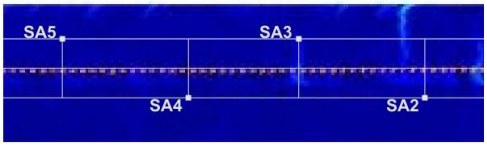


Figure 128, Beam S10H2A, (a) Group location on the beam, (b)-(h) DIC observations

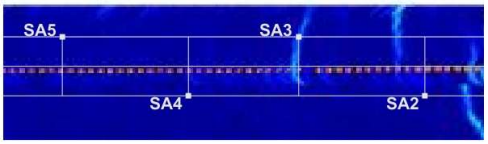
(b)- 300[kN]



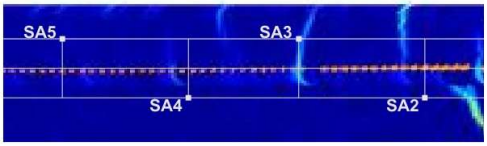
(c)- 650[kN]



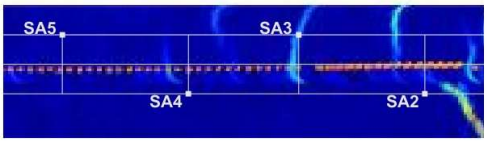
(d)- 700[kN]



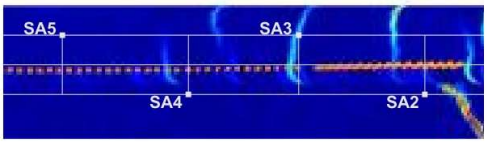
(e)- 750[kN]



(f)- 800[kN]



(g)- 850[kN]



(h)- 900[kN]

4.3.1 – Beam S10H1D

Accuracy Assessment

Figure 129 presents the evolution of the NRMSD values with increasing load. From this graph, two distinct phases can be identified. Between load levels of 0 and 850 kN, the NRMSD values remain low and relatively stable, indicating that the tomographic reconstruction of the ultrasonic data is reliable and the reconstructed fields can be confidently interpreted. After 850kN, the NRMSD values begin to increase, suggesting a gradual reduction in reconstruction accuracy, indicating a substantial loss of reliability in the tomographic reconstruction. Consequently, the results obtained after this load level are not further examined.

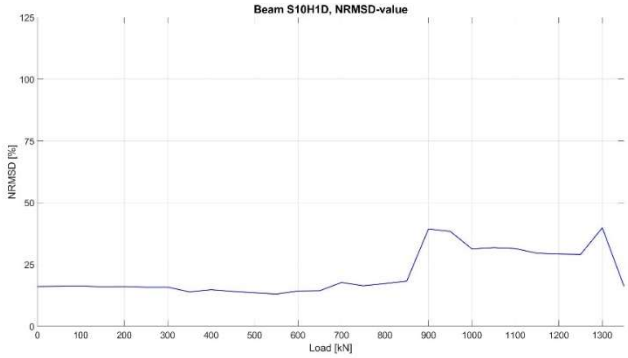


Figure 129, normalised RMSD-value graph of beam S10H1D

Analysis of the Reconstructed Tomographic Indicators

For the tomography analysis of beam S10H1D, Group 1, the Graph presented below identify POI at load levels 600 kN, 650 kN, 800 kN and 900 kN. At 600 kN, Fields I and II decrease below 0.80, which may be associated with multiple cracks observed within these regions. At 650 kN, a pronounced indicator valley is observed across Fields III–VI, consistent with multiple cracks observed in these regions. At 800 kN, Fields I and II again exhibit a valley, while cracks become more pronounced. At 900 kN, all monitored fields decrease below 0.80, consistent with widespread cracking.

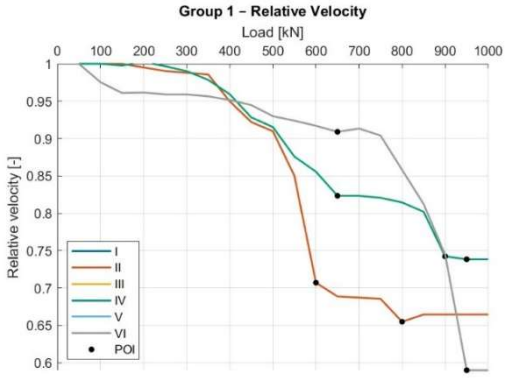


Figure 130, Beam S10H1D Group 1 RV-overview Graph

At 600 kN, no POI were identified for Fields III–VI, although their values range between 0.85 and 0.90. The observations, however, indicate similar crack development in these regions as in Fields I and II, whose relative velocity values decrease to approximately 0.70.

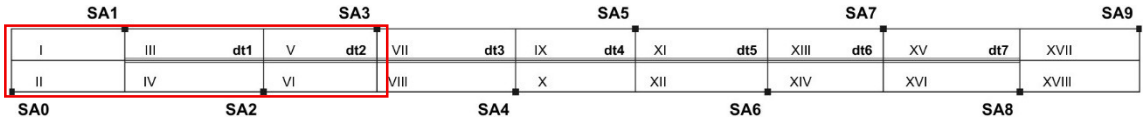
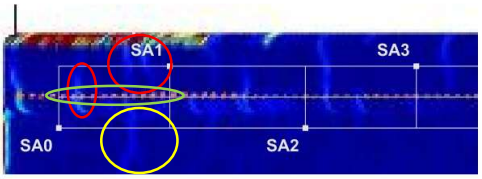
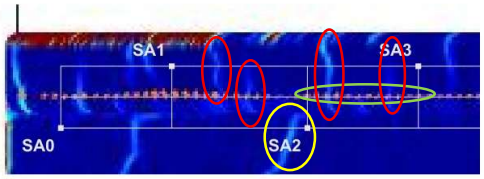


Figure 131, Beam S10H1D, (a) Group location on the beam, (b)-(c) DIC observations



(b)- 600[kN]



(c)- 650[kN]

For Group 2, POI are identified at load levels 650 kN, 700 kN, 750 kN and 850 kN. At 650 kN, Fields V and VI exhibit a valley, while multiple cracks are observed within the structural region. At 700 kN, Fields IX and X exhibit a valley. At 750 kN, Fields VII and VIII decrease below 0.80, consistent with flexural cracks previously observed in these regions. At 850 kN, Fields V–VIII decrease below 0.80, while cracks become more pronounced.

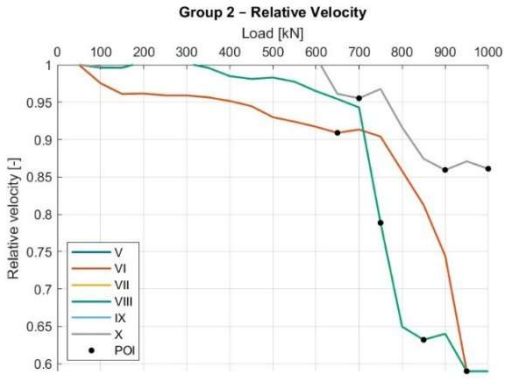
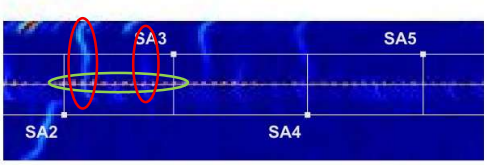


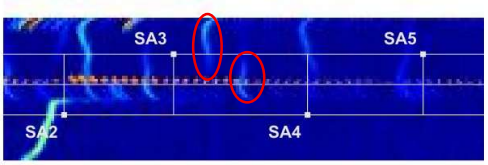
Figure 132, Beam S10H1D Group 2 RV-overview Graph

At 650 kN, several cracks are already present within the monitoring region of this group while the relative field velocity values remain above 0.90. The indicator values only begin to decrease after approximately 750 kN, even though cracks are already clearly visible at earlier load levels.

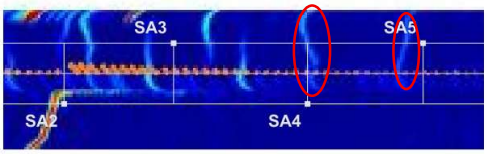
Figure 133, Beam S10H1D, (a) Group location on the beam, (b)-(d) DIC observations



(b)- 650[kN]



(c)- 750[kN]



(d)- 900[kN]

Analysis of the Interface Time Interference

Considering the tomography region using dt1–dt3 for beam S10H1D, POI occur at 400 kN, 550 kN, 600 kN, 700 kN, 800 kN, 950 kN and 1000 kN. At 400 kN, all delay-time indicators remain below 1.0. At 550 kN, dt1 approaches 2.0 while dt2 and dt3 remain near 1.0; the DIC observations show signs of delamination in the dt1 region. At 600 kN and 700 kN, dt2 increases substantially to above 2.0 while dt1 increases to approximately 3.0, consistent with progressive delamination, as the reference observations show damage extending from the dt1 region towards dt2 and approaching dt3. At 800 kN and 950 kN, the dt indicators decrease while delamination becomes more pronounced. At 1000 kN, dt3 exceeds 2.0 while dt1 and dt2 remain around 3.0, with delamination becoming increasingly pronounced.

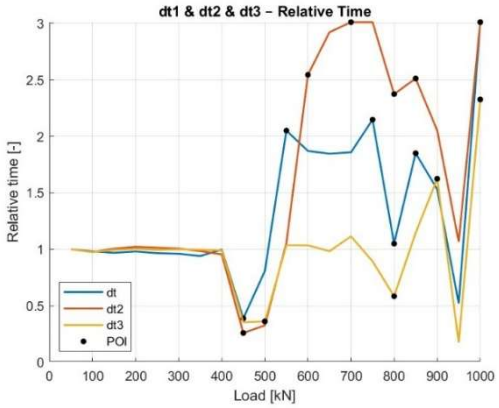


Figure 134, Beam S10H1D dt_1-dt_3 overview Graph

A noticeable dip in the dt values occurs at 450 kN and 500 kN and a second sudden decrease is observed in all dt indicators at 950 kN, which may indicate a calculation issue.
A dt value near 2.0 appears to correspond with the presence of delamination, as observed at 550 kN. However, dt3 does not reach this level until approximately 900 kN, while the DIC observations indicate clear delamination in this region as early as 750 kN

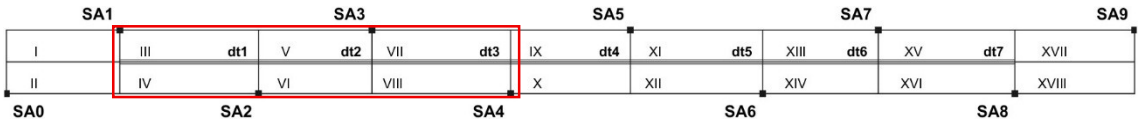
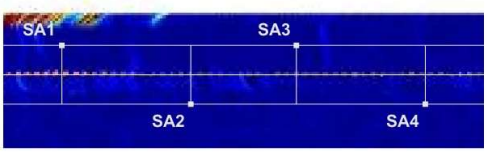
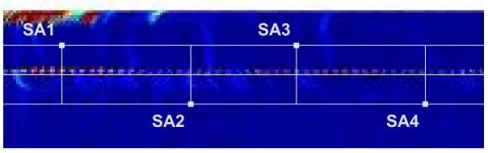


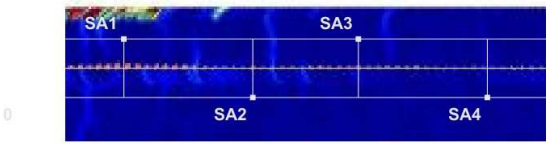
Figure 135, Beam S10H1D, (a) Group location on the beam, (b)-(j) DIC observations



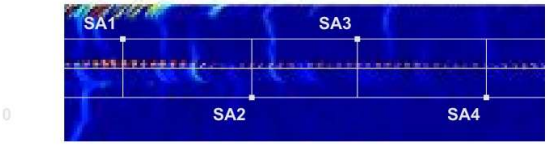
(b)- 450[kN]



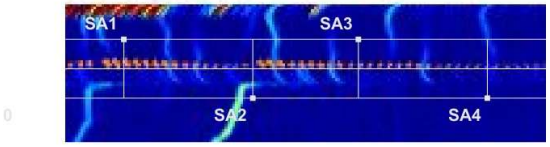
(c)- 500[kN]



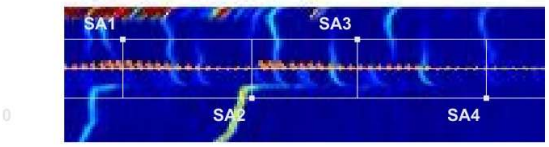
(d)- 550[kN]



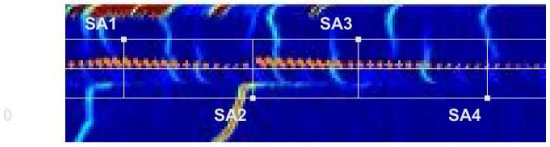
(e)- 600[kN]



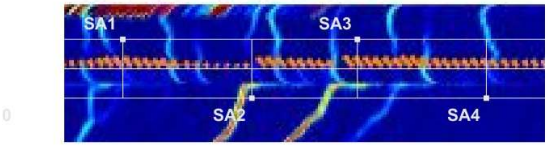
(f)- 750[kN]



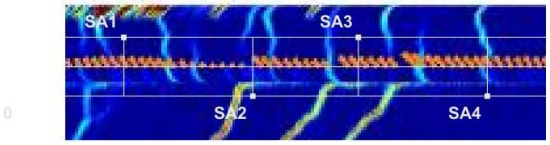
(g)- 800[kN]



(h)- 850[kN]



(i)- 950[kN]



(j)- 1000[kN]

Finally, for the tomography region using dt3–dt5 in beam S10H1D, POI are identified at 500 kN, 750 kN, 800 kN, 900 kN and 1000 kN. At 500 kN, all indicators remain below 1.0 without clear signs of internal delamination. At 750 kN, dt4 and dt5 increase to approximately 1.5 and 2.0 respectively, while dt3 remains near 1.0; reference observations show delamination in the dt3 region. At 800 kN, all indicators decrease while the DIC observations show delamination only in the dt3 region. At 900 kN, dt3 and dt4 increase to approximately 1.5 while dt5 reaches around 3.0, with delamination still limited to the dt3 region. At 1000 kN, dt3 and dt4 increase further to approximately 2.0 and 3.0 respectively while dt5 remains around 3.0. At this load level, the reference observations show delamination extending across the entire region from dt3 to dt5.

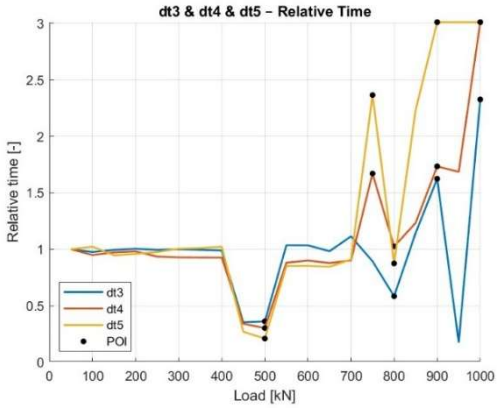
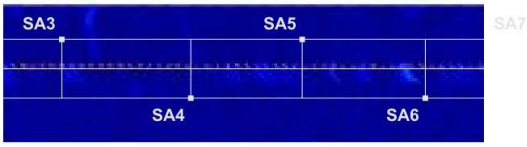


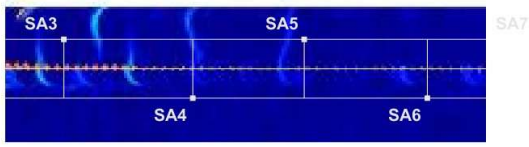
Figure 136, Beam S10H1D dt_3-dt_5 overview Graph

The dt indicators do not increase smoothly but instead fluctuate between load steps. The values also do not consistently remain above 2.0, making it difficult to assess whether this value can be considered a reliable threshold for identifying delamination. This behaviour becomes more pronounced at higher load levels, which may again be influenced by structural deformation affecting the tomographic calculations.

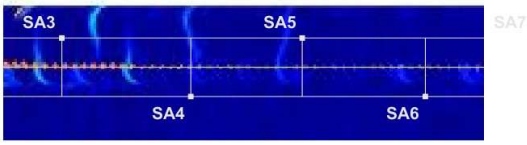
Figure 137, Beam S10H1D, (a) Group location on the beam, (b)-(g) DIC observations



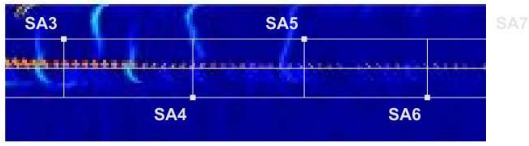
(b)- 500[kN]



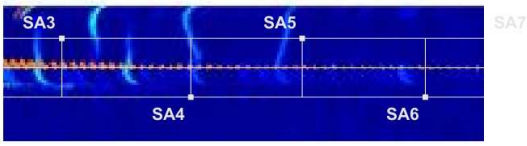
(c)- 750[kN]



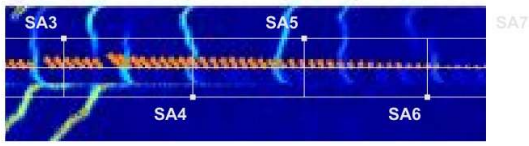
(d)- 800[kN]



(e)- 850[kN]



(f)- 900[kN]



(g)- 1000[kN]

Chapter 5 – Evaluation and Discussion of the Representation of the Ultrasonic Indicators

This chapter interprets the experimental results presented in Chapter 4 in relation to the research sub-questions of this study, with particular emphasis on how the results are represented in the two graph types: the Ray Path graphs and the tomographic plots. The performance of the ultrasonic indicators are evaluated with respect to three aspects: (1) early detection of structural changes prior to visible cracking, (2) localisation of crack formation within the girder, and (3) representation of crack magnitude and progression.

This chapter is structured as follows. Sub-chapter 5.1 examines the Ray Path graphs, focusing on how Coda Wave Interferometry (CWI) and Ultrasonic Pulse Velocity (UPV) indicators are represented and how effectively they capture crack initiation, localisation, and development. Sub-chapter 5.2 evaluates the tomographic results, discussing the behaviour of the field velocity and *dt* indicators and their ability to represent spatial damage patterns and interface disturbances. Further in this chapter, the combined approach introduced earlier is examined, focusing on the complementary use of CWI-based indicators for early detection of anomalies and UPV-derived relative velocity for spatial localisation of cracking and interface-related damage. The key observations and discussion points are presented in Sub-chapter 5.4, highlighting the strengths and limitations of the applied ultrasonic monitoring approach.

A complete overview of the comparison between each ultrasonic indicator and the corresponding DIC observations for all available load steps is provided in Appendix V – Ultrasonic Graphs Analysis. In this chapter, selected load steps for each beam are examined in detail, where key observations and discussion points can be drawn; for each case, three consecutive load steps are presented to enable assessment of the progression in the ultrasonic graphs.

5.1 – Evaluation of the Ray Path Representation

Ray Path visualisations represent the ultrasonic response along the direct propagation paths between Smart Aggregate sensor pairs. This approach provides an intuitive means of relating changes in the ultrasonic indicators to the geometry of the beam and the location of the sensors.

5.1.1 – Representation of CWI Indicators in Ray Path Graphs

A key objective of this study was to determine whether CWI indicators could detect structural changes before cracks became visible in the Digital Image Correlation (DIC) measurements, with a focus on the onset of cracking. The results presented in Chapter 4 show that variations in both ε and CC occurred at load levels where no visible cracks were present; however, the CC indicator more consistently corresponded with the onset of observable cracking.

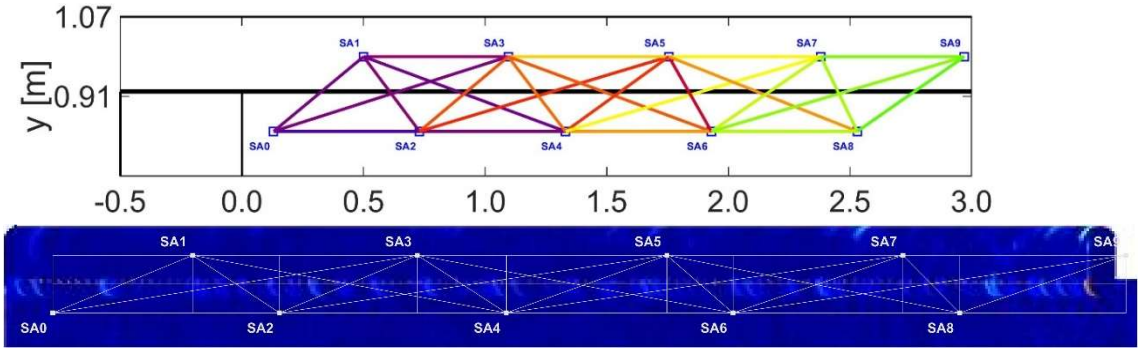
While CWI confirms that structural disturbance occurs within the monitored region, localisation of cracks requires a more precise examination of the ultrasonic response along individual SA pairs. In this study, UPV measurements were analysed to evaluate how effectively the ultrasonic indicators could identify the location and type of cracking based on the orientation of the sensor pairs. However, some degree of localisation could still be obtained by comparing the response of SA pairs within the same group using the CWI indicators.

For the CWI indicators, the selection of an appropriate threshold value requires careful consideration. A commonly referenced threshold of $\varepsilon < -2.0e{-}4$ was evaluated; however, its reliability proved inconsistent across the analysed beams. In contrast, the correlation coefficient (CC) provided a more stable reference, particularly when considering reductions below 0.80. This also affects the applicability of the $CC < 0.80$ condition, frequently rendering the ε results unreliable and resulting in ε-graphs containing multiple undefined regions.

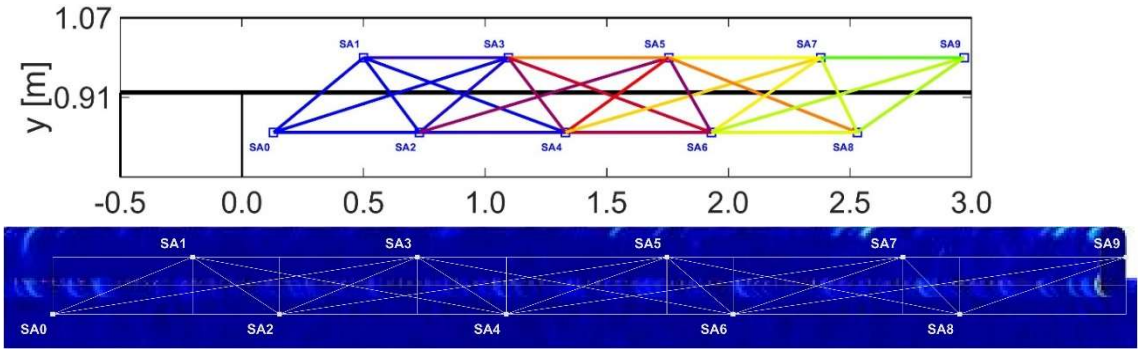
Early detection using relative velocity change

For several SA groups, the relative velocity change exceeded the commonly referenced threshold of ε<−2.0e−4 before cracks were observed in the DIC measurements. This suggests that coda waves are sensitive to early structural disturbances occurring prior to visible crack formation. Plotting the ε-value in a Ray Path graph highlights this phenomenon better, in the Figure 138, Figure 139 and Figure 140 the early stages (250-350 kN) of all three beams can be seen with there corresponding DIC observations.

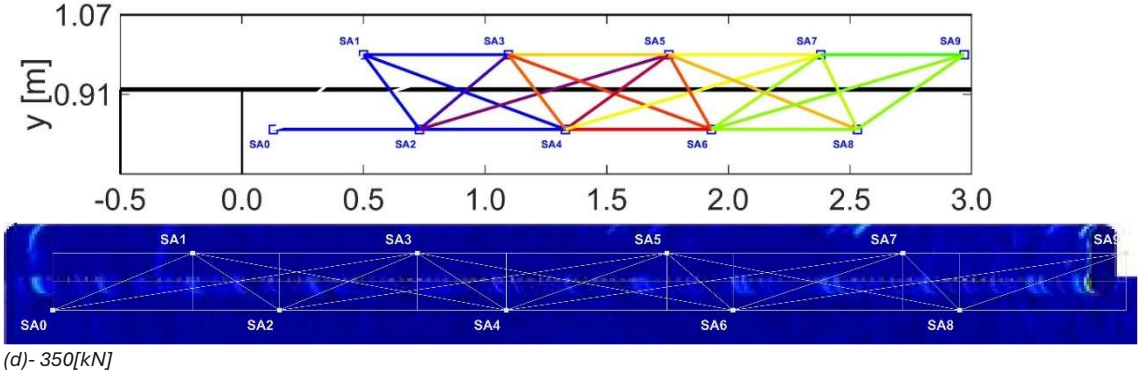
Figure 138, Early load stages (250-350 kN) ε-Graph of Beam S10H2D, (a)- colour bar of ε-value, (b)-(d) ε-Graph v DIC observations



(b)- 250[kN]

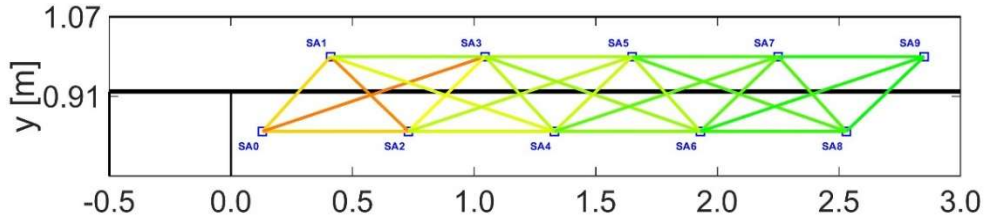


(c)- 300[kN]



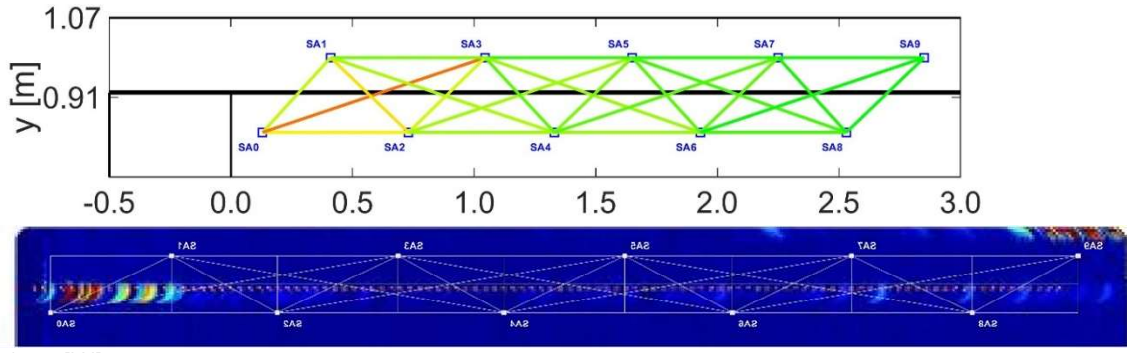
(d)- 350[kN]

Figure 139, Early load stages (250-350 kN) ε-Graph of Beam S10H2A, (a)- colour bar of ε-value, (b)-(d) ε-Graph v DIC observations (Flipped over y-axis)

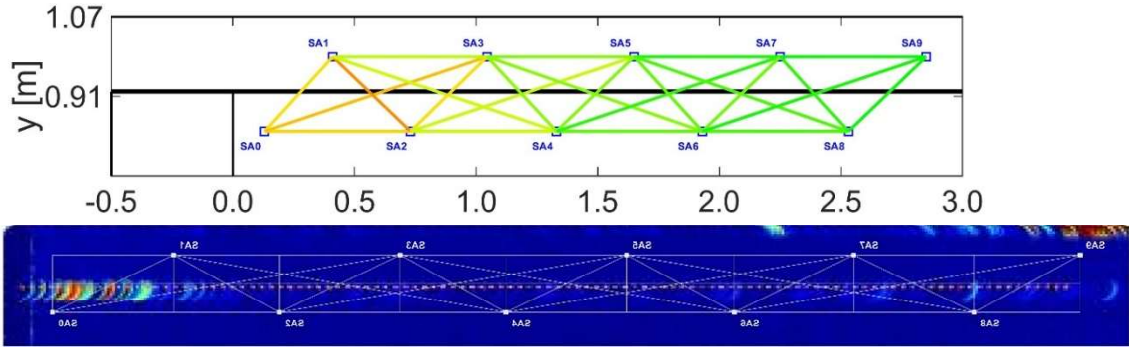


(no DIC figure)

(b)- 250[kN]

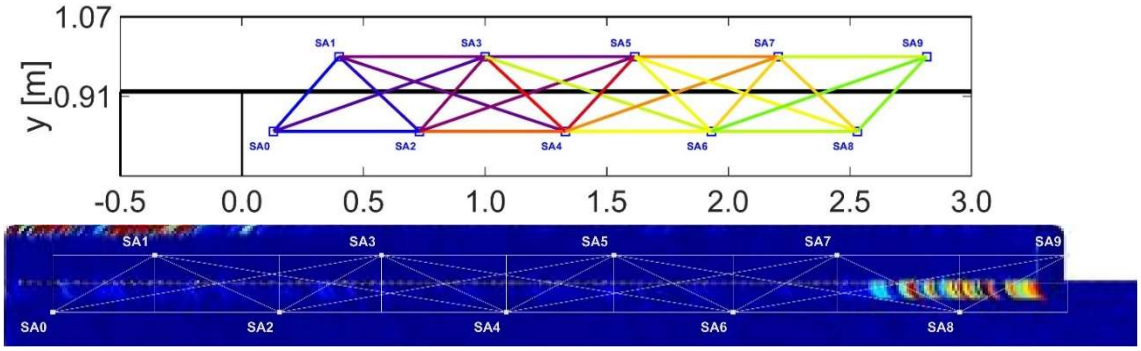


(c)- 300[kN]

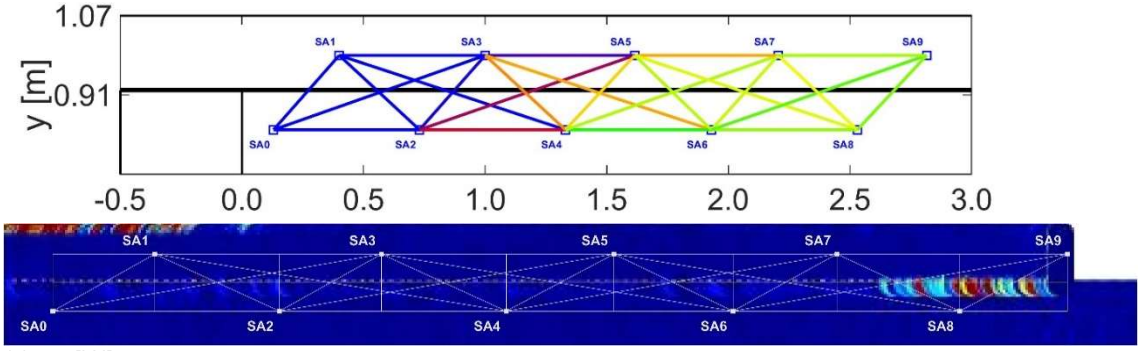


(d)- 350[kN]

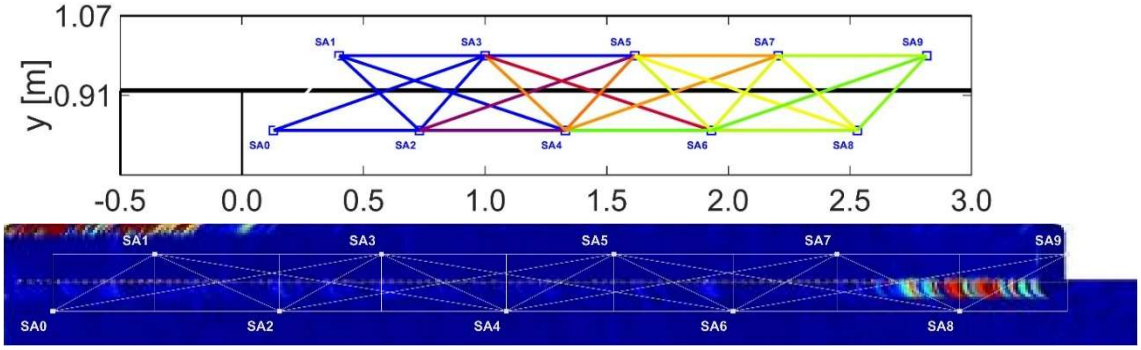
Figure 140, Early load stages (250-350 kN) ε-Graph of Beam S10H1D, (a)- colour bar of ε-value, (b)-(d) ε-Graph v DIC observations



(b)- 250[kN]



(c)- 300[kN]



(d)- 350[kN]

It can be observed that for beams S10H2D and S10H1D, the threshold value is reached before cracks appear. For beam S10H2A, this is not observed, as crack initiation occurs later and develops more rapidly (see Appendix V – Ultrasonic Graphs Analysis).

These observations indicate that ε cannot be considered a standalone early crack indicator, as the results demonstrate that additional disturbances influence the ε-values prior to visible cracking. These disturbances may be associated with microcracking, local stress redistribution, or other stress-induced changes affecting ultrasonic wave propagation.

Furthermore, the behaviour of the ε-values across the Ray Path graphs reflects the influence of stress redistribution within the beam. Since ε is sensitive to small variations in wave velocity caused by stress changes, gradual variations can be observed across successive load steps. As the applied load increases, the evolving stress state influences ultrasonic wave propagation, which appears in the Ray Path graphs as a gradual shift in colour across multiple sensor paths

with increasing load levels. This behaviour is most clearly observed over the full loading progression, as presented in Appendix V – Ultrasonic Graphs Analysis.

Early detection using Correlation Coefficient

Focusing on early detection, the CC-values remain relatively stable during the initial loading stages and begin to decrease once cracking initiates or propagates within the monitored region. Compared to ε, this response occurs at a slightly later stage but shows a clearer correspondence with observable cracking.

Several CC-overview graphs illustrate this behaviour, which continues into the later load stages. As the CC value represents the correlation with the signal from a previous load step, it effectively captures changes in wave propagation conditions between successive loading stages. In Figure 141, Figure 142 and Figure 143, the initial crack development stages of all three beams are shown together with the corresponding DIC observations, noting that different load stages are highlighted for each beam.

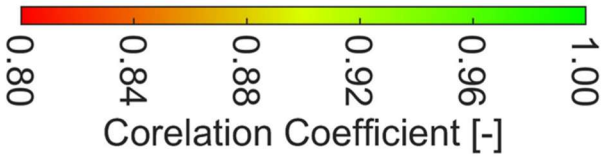
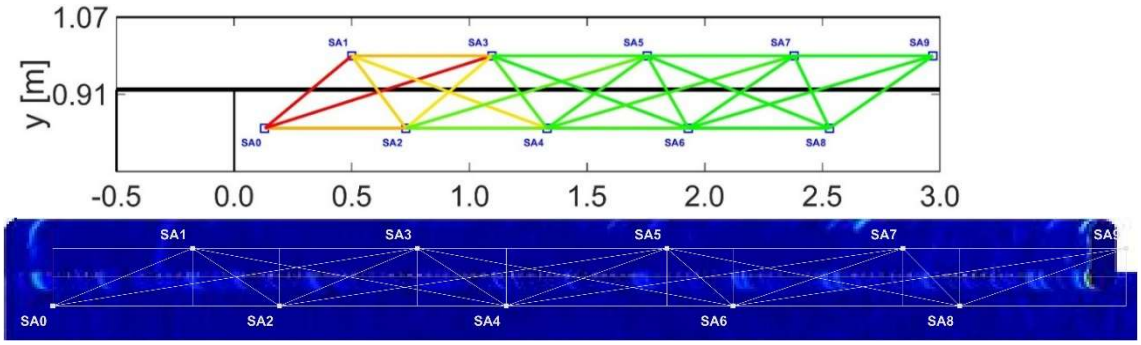
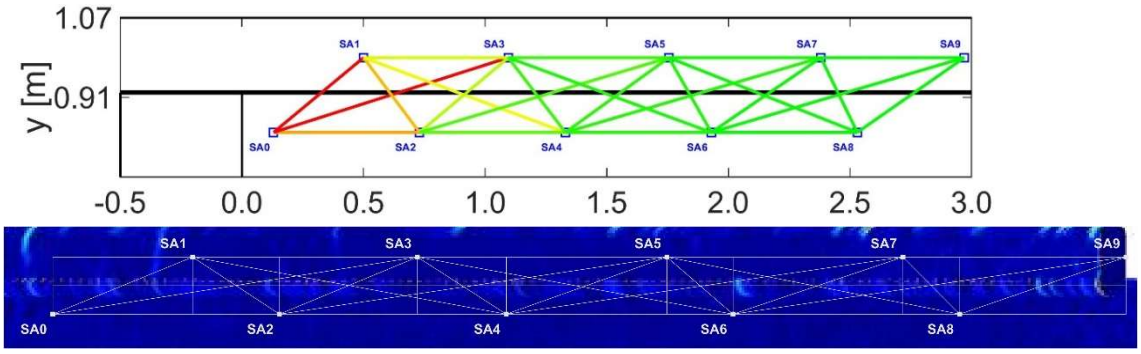


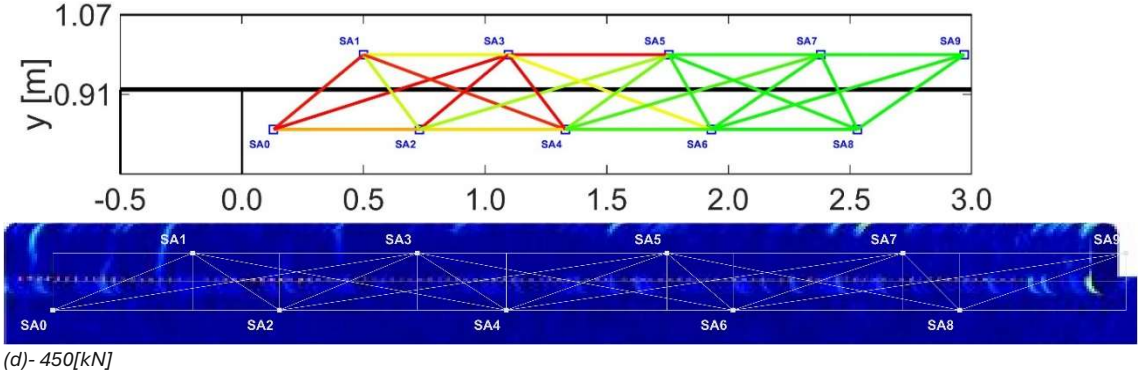
Figure 141, Initial crack development stages (350-450 kN) CC-Graph of Beam S10H2D, (a)- colour bar of CC-value, (b)-(d) CC-Graph v DIC observations S10H2D



(b)- 350[kN]



(c)- 400[kN]



(d)- 450[kN]

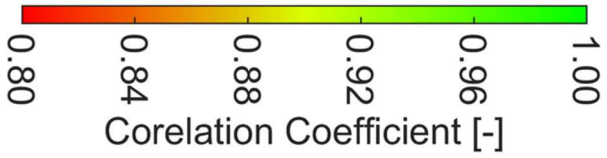
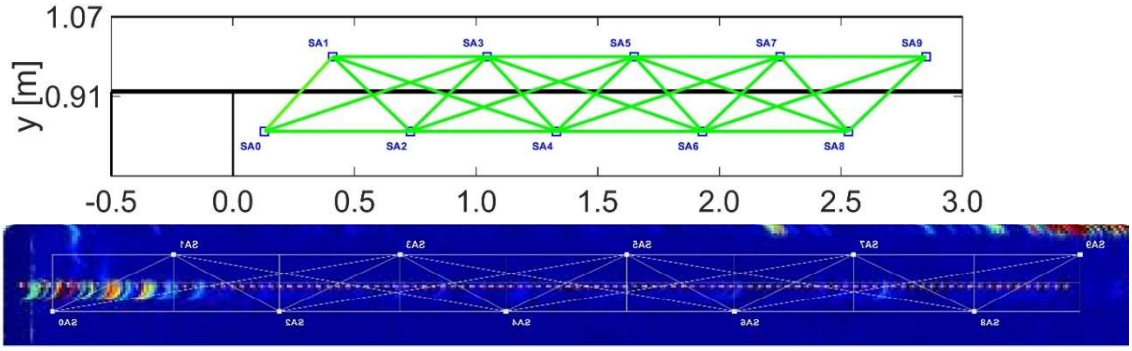
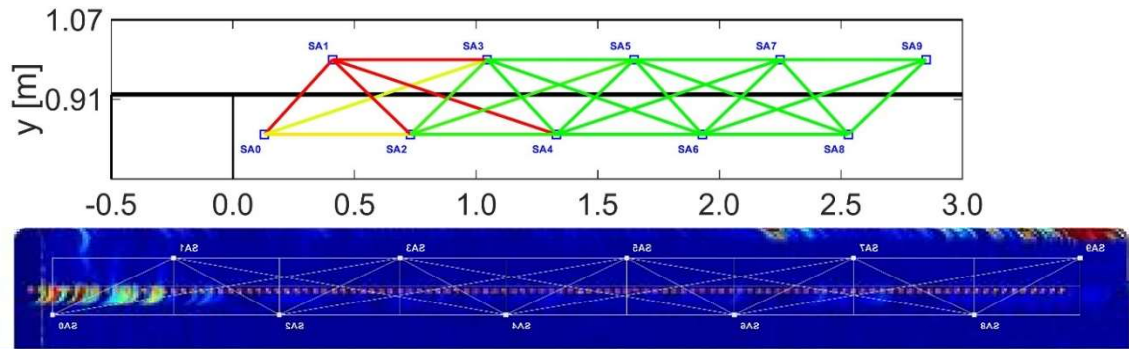


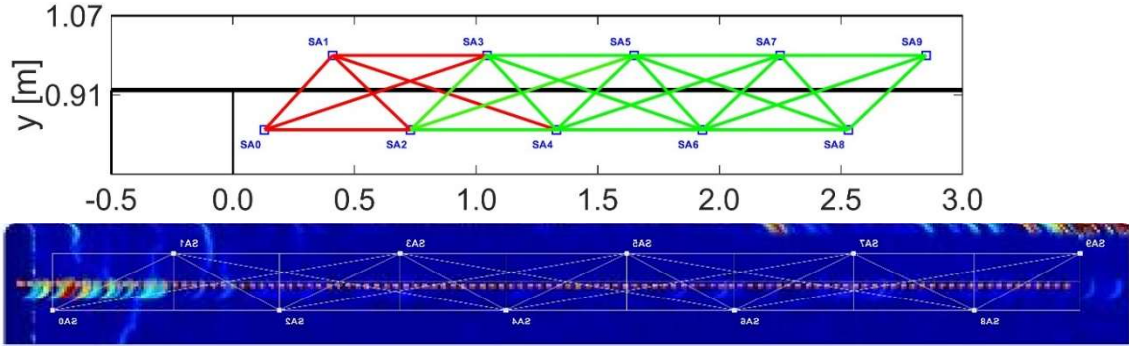
Figure 142, Initial crack development load stages (550-650 kN) CC-Graph of Beam S10H2A, (a)- colour bar of CC-value, (b)-(d) CC-Graph v DIC observations (Flipped over y-axis)



(b)- 550[kN]



(c)- 600[kN]



(d)- 650[kN]

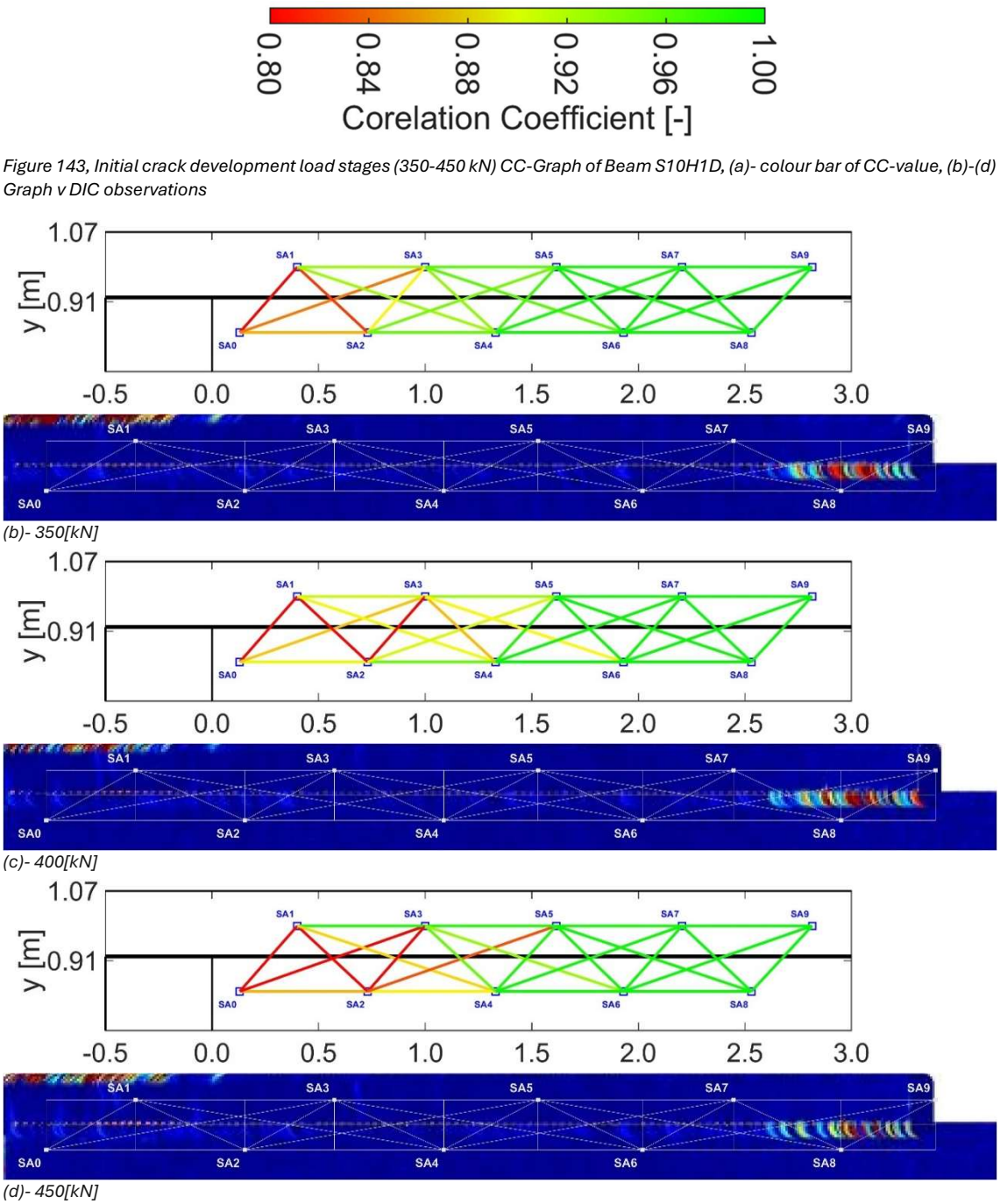


Figure 143, Initial crack development load stages (350-450 kN) CC-Graph of Beam S10H1D, (a)- colour bar of CC-value, (b)-(d) CC-Graph v DIC observations

(b)- 350[kN]

(c)- 400[kN]

(d)- 450[kN]

When the CC indicators are displayed in Ray Path graphs, a simplification is introduced. CWI measurements reflect the response of a distributed wave field interacting with a relatively large portion of the monitored region, whereas the Ray Path visualisation represents this response as a single line between two sensors. As a result, part of the spatial complexity is condensed into a single propagation path.

Despite this simplification, a degree of spatial distribution can still be identified. When a disturbance develops within a specific region, the ray paths intersecting that region generally show stronger indicator changes, while neighbouring paths show progressively smaller changes. In the Ray Path graphs, this appears as clusters of red paths near the disturbance, gradually transitioning to greener paths further away. This pattern therefore provides a qualitative indication of both the location and spatial influence of the disturbance.

With respect to threshold behaviour, a reduction in CC generally coincides with the initiation or early propagation of cracks observed in the DIC measurements. Based on the area analysis presented in Chapter 4, it is not a fixed threshold but rather the relative drop in CC that most clearly indicates the onset of structural disturbance. This reflects the increasing decorrelation of ultrasonic waveforms as cracking alters the propagation medium. Consequently, the CC indicator provides a consistent qualitative indication of structural changes within the monitored region.

5.1.2 – Representation of UPV Indicator in Ray Path Graphs

In this study, UPV measurements were primarily analysed to evaluate how effectively the ultrasonic indicators could identify the location and type of cracking based on the category of the sensor pairs. The SA pairs were therefore interpreted according to their relation to potential crack mechanisms, including flexural cracking, shear cracking, and possible interface-related delamination.

Localisation through Relative Velocity

For the UPV indicator, the Ray Path representation corresponds directly to the measurement principle, as the indicator is derived from the arrival time of the ultrasonic pulse travelling between the two sensors. In Figure 144, Figure 145 and Figure 146, note the varying load stages for each beam; these figures were selected to highlight the onset of changes in the relative velocity values.

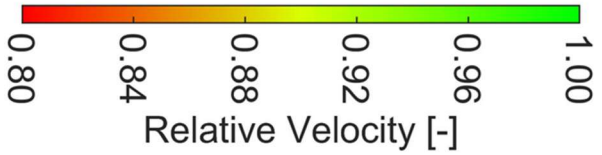
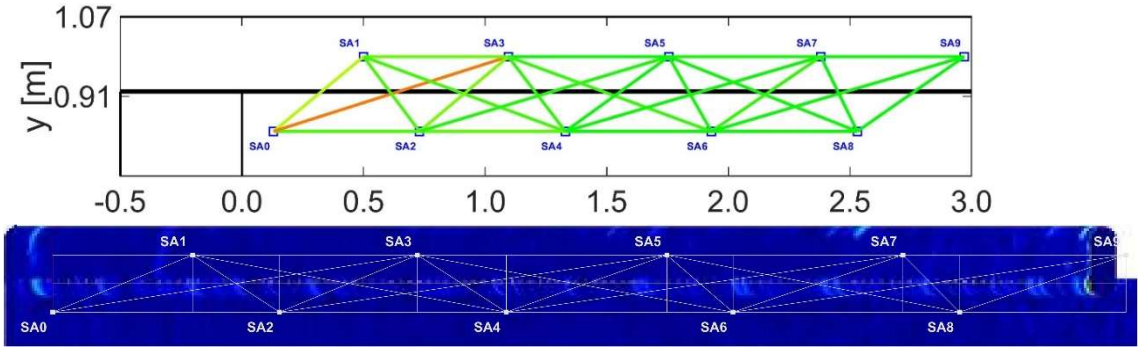
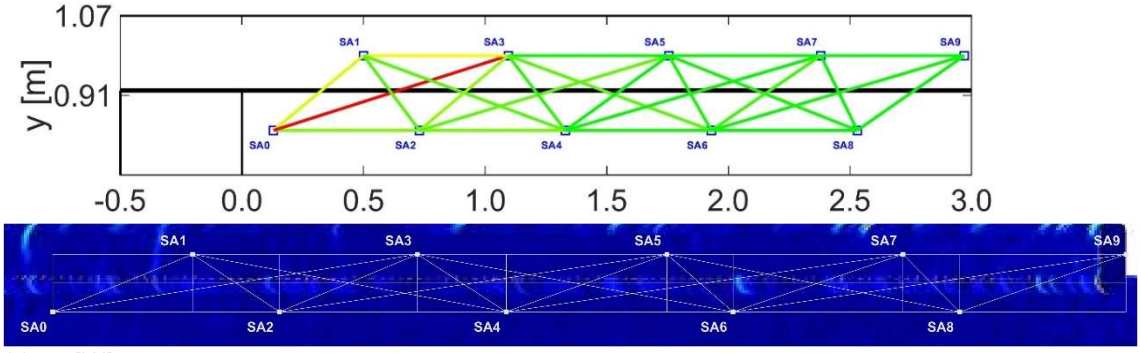


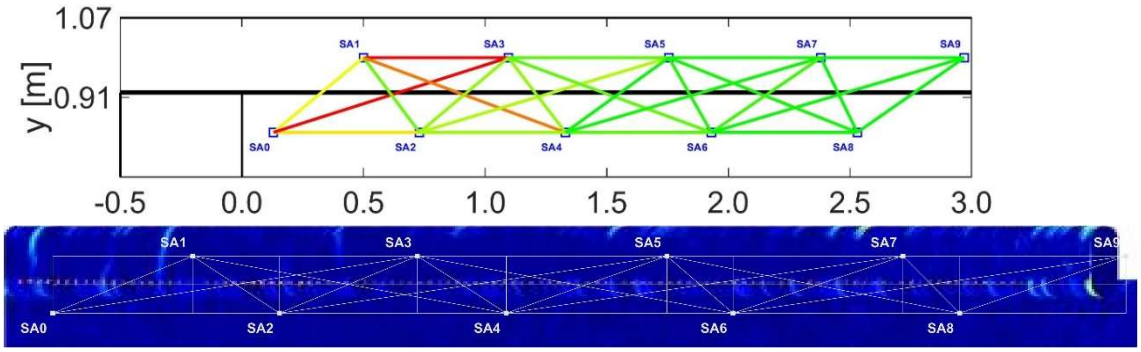
Figure 144, Onset load stages (350-450 kN) RV-Graph of Beam S10H2D, (a)- colour bar of RV-value, (b)-(d) RV-Graph v DIC observations



(b)- 350[kN]



(c)- 400[kN]



(d)- 450[kN]

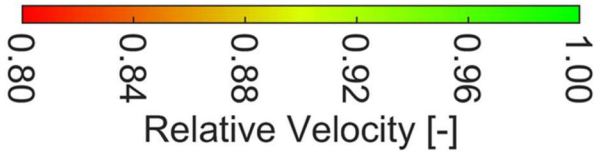
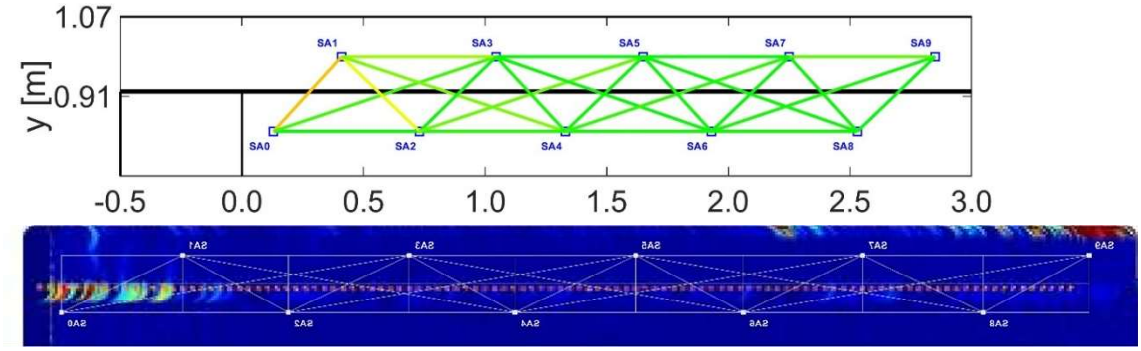
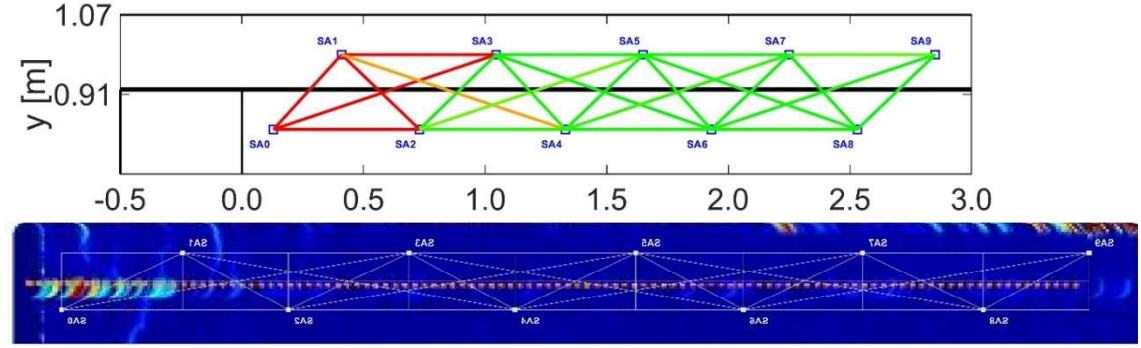


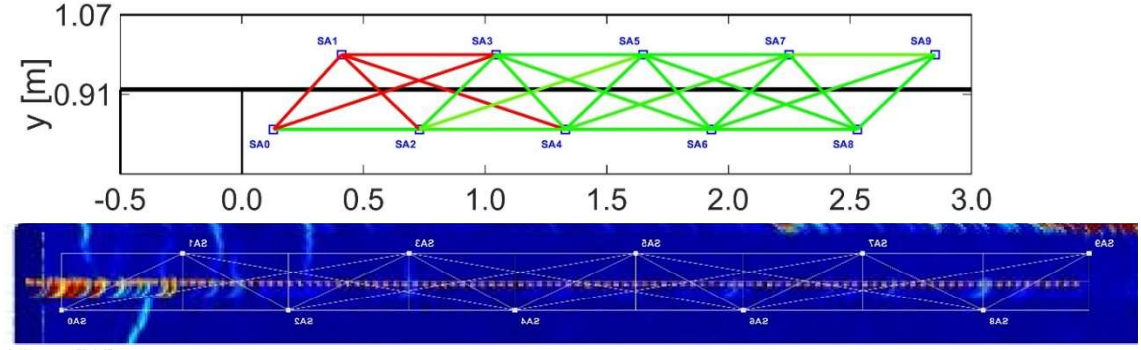
Figure 145, Onset load stages (600-700 kN) RV-Graph of Beam S10H2A, (a)- colour bar of RV-value, (b)-(d) RV-Graph v DIC observations (Flipped over y-axis)



(b)- 600[kN]



(c)- 650[kN]



(d)- 700[kN]

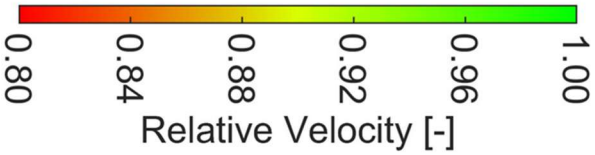
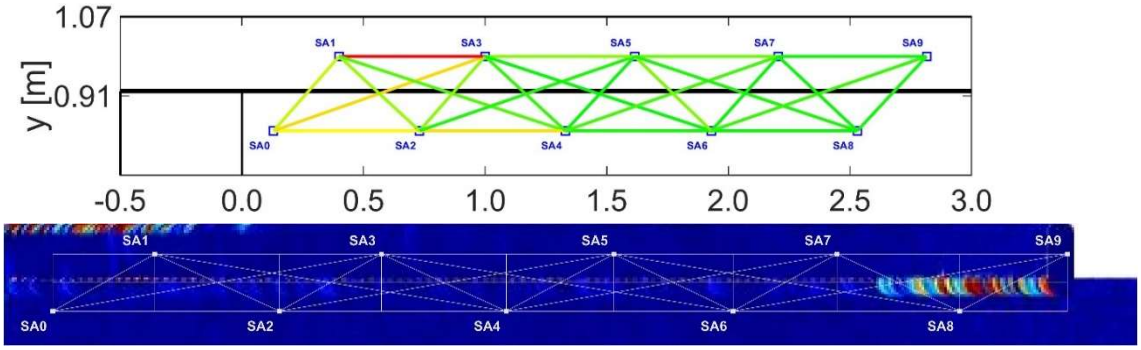
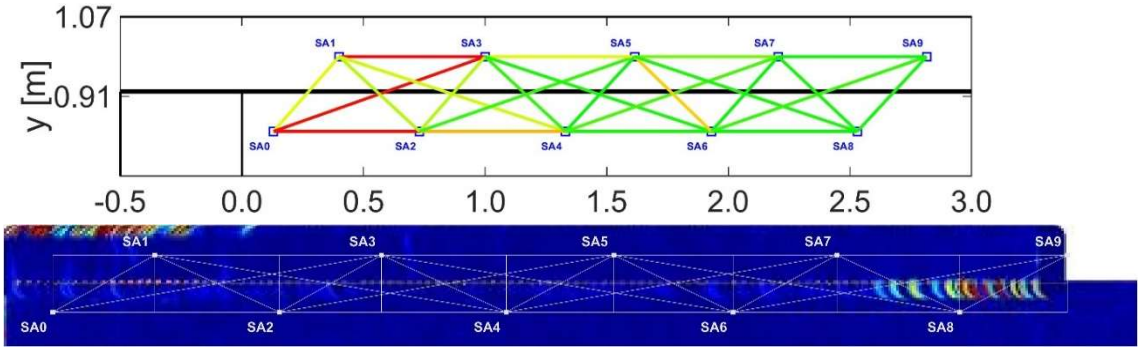


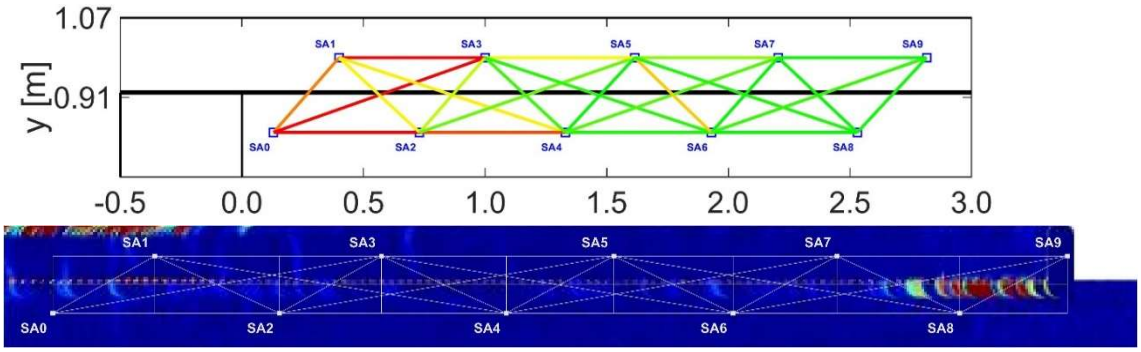
Figure 146, Onset load stages (400-500 kN) RV-Graph of Beam S10H1D, (a)- colour bar of RV-value, (b)-(d) RV-Graph v DIC observations



(b)- 400[kN]



(c)- 450[kN]



(d)- 500[kN]

The effectiveness of crack localisation depends strongly on the orientation of the SA pairs relative to the crack patterns. The horizontal top and bottom SA pairs generally produce clearer and more interpretable responses than the diagonal pairs. Reductions in relative velocity along these paths correspond well with the development of flexural and shear cracks observed in the beams. Interpretation becomes more challenging when a single ray path intersects multiple cracks, as the measured arrival time reflects the combined influence of multiple disturbances along the path, making it difficult to attribute changes to a specific crack. This applies both to the presence of multiple cracks and to different crack types, the latter being particularly relevant for diagonal SA pairs.

In the Figure 147, Figure 148 and Figure 149 illustrate the relative velocity response at later loading stages.

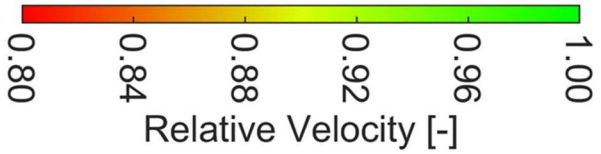
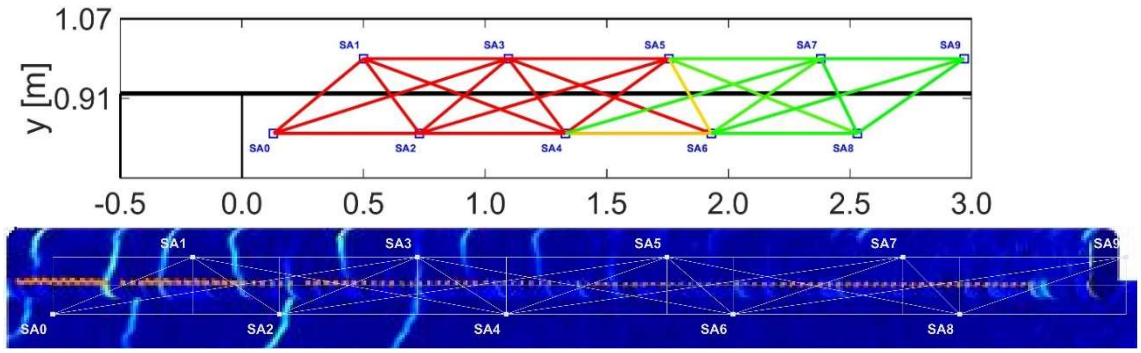
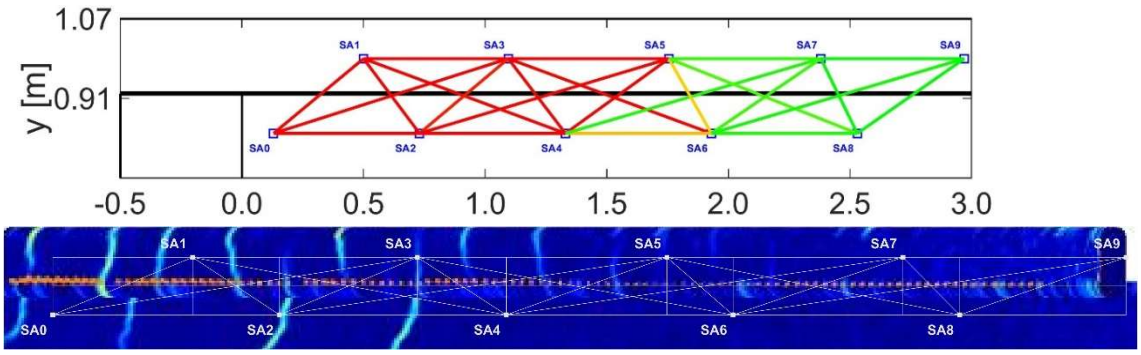


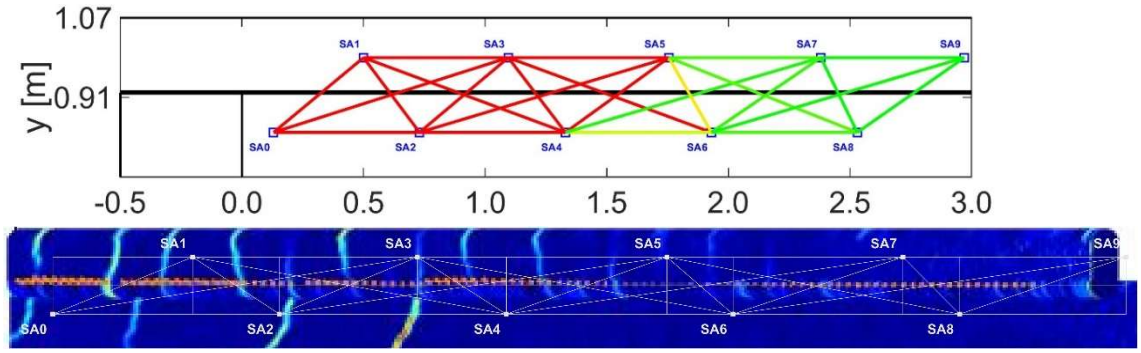
Figure 147, Later load (800-900 kN) stages RV-Graph of Beam S10H2D, (a)- colour bar of RV-value, (b)-(d) RV-Graph v DIC observations



(b)- 800[kN]



(c)- 850[kN]



(d)- 900[kN]

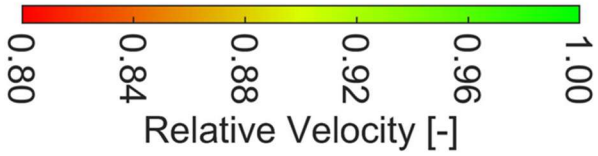
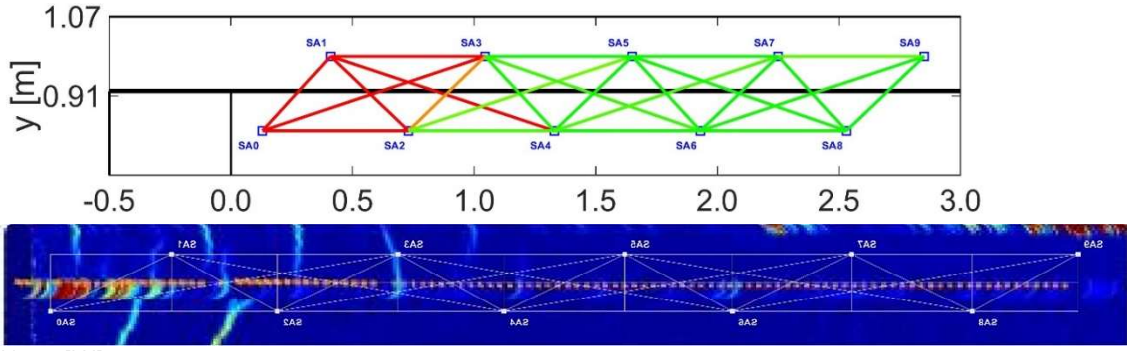
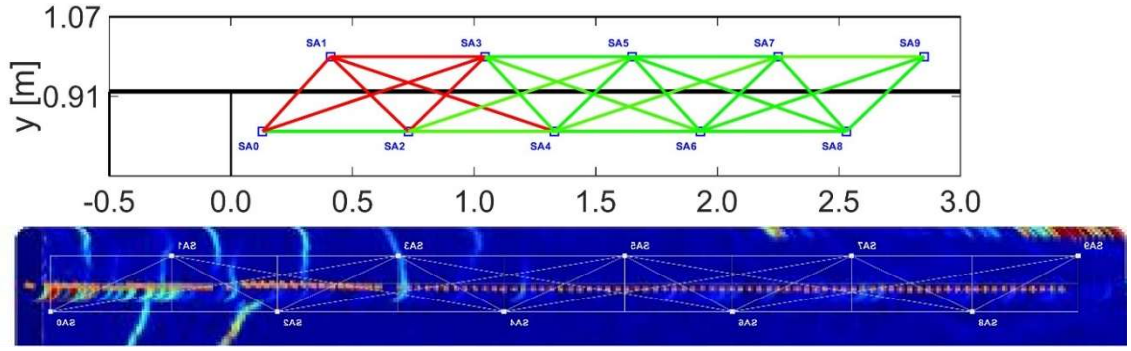


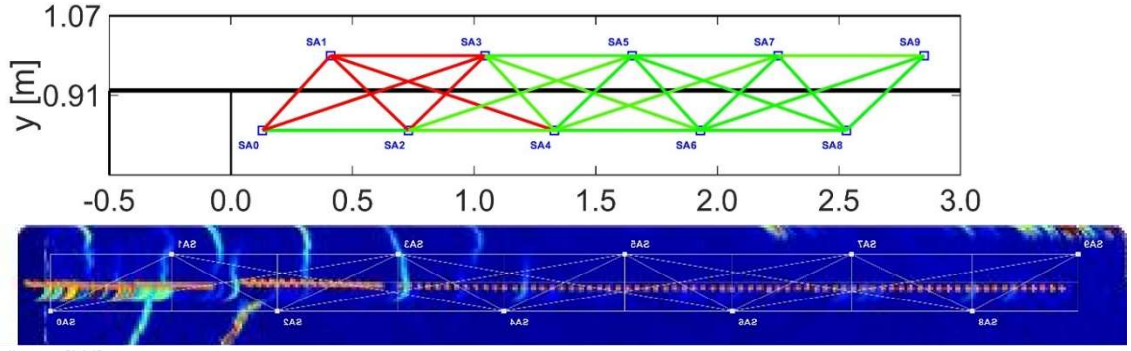
Figure 148, Later load stages (800-900 kN) RV-Graph of Beam S10H2A, (a)- colour bar of RV-value, (b)-(d) RV-Graph v DIC observations (Flipped over y-axis)



(b)- 800[kN]



(c)- 850[kN]



(d)- 900[kN]

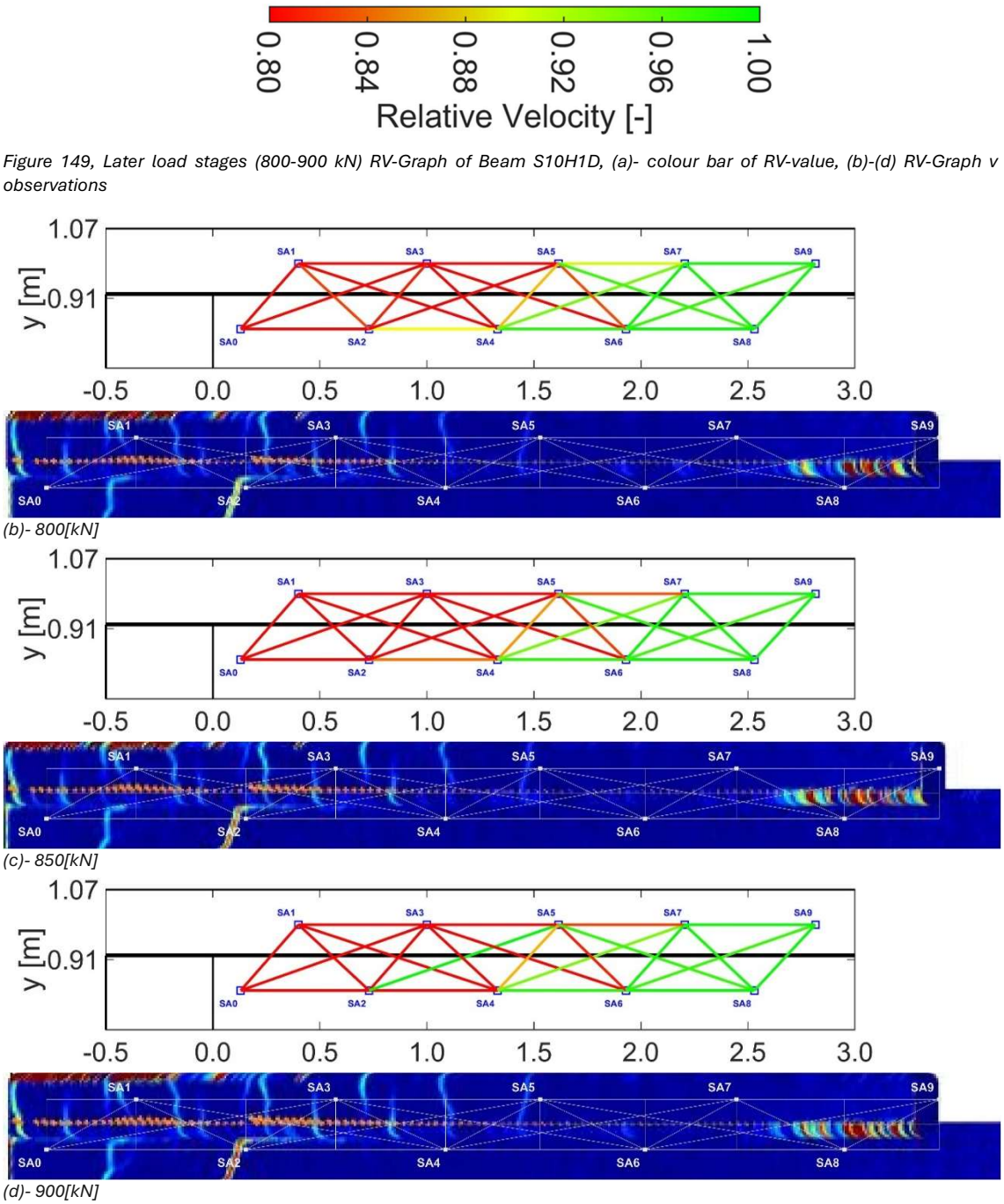


Figure 149, Later load stages (800-900 kN) RV-Graph of Beam S10H1D, (a)- colour bar of RV-value, (b)-(d) RV-Graph v DIC observations

(b)- 800[kN]

(c)- 850[kN]

(d)- 900[kN]

The DIC observations of beams S10H2D and S10H2A (Figure 147 and Figure 148) indicate clear delamination along a large portion of the beam length. However, the diagonal SA pairs in both beams do not consistently reflect this behaviour. From the Ray Path analysis in Chapter 4, it was observed that several diagonal SA pairs do not respond as directly to interface delamination. As introduced in Chapter 2, the orientation of the SA pairs and their corresponding ray paths plays a governing role in the sensitivity of the measurement. In the case of diagonal paths, the propagation path intersects the interface under an oblique angle and over a longer distance, causing the influence of delamination to be distributed and combined with other disturbances along the path. As a result, the specific contribution of interface separation is less pronounced in the relative velocity response.

The location of the SA groups along the beam also significantly influences the interpretation of the results. Group 1, located closer to the hogging moment region, experiences a greater number and variety of cracks, increasing the likelihood that individual ray paths intersect multiple cracks. This results in more complex ultrasonic responses. In contrast, groups located further away generally experience fewer cracks, leading to clearer results.
Additionally, Figure 149 demonstrate that the top and bottom SA pairs continue to provide consistent indications of crack development. However, Figure 148 highlights a limitation: for beams S10H2A SA pairs 0—2 and 2—4, green Ray Paths are observed despite clear cracking in the DIC results, which is likely attributable to signal loss. And for beam S10H1D, the top and bottom SA pairs do not line up with the DIC observations.

Threshold and Continued Reduction in Relative Velocity

The interpretation of UPV indicators first requires evaluation of the threshold value, followed by analysis of the behaviour beyond this point. The relative velocity graphs were constructed using a threshold value of 0.80 where clear crack development was observed for the top and bottom SA pairs at this level. Once the relative velocity reaches this threshold, further reductions could be associated with continued structural development.
For diagonal SA pairs, however, the magnitude of the velocity reduction cannot be directly related to a specific crack. Instead, the UPV indicators should be interpreted as relative measures of increasing disturbance along the propagation path, reflecting the cumulative effects of crack development within the monitored region.
Continued reductions in relative velocity may result from crack widening, crack extension, or the formation of additional cracks intersecting the ray path. This behaviour was observed for several top and bottom category SA pairs, where progressive reductions corresponded with continued crack development observed in the DIC measurements, particularly after the threshold value had been reached. However, these additional reductions cannot be directly correlated to a specific crack width or crack length.

5.2 – Evaluation of Tomography Representation

Tomography was used to visualise spatial variations in ultrasonic response within the monitored region. In this study, two tomographic variables were evaluated: the reconstructed field velocity, derived from the velocity-based tomography, and the interface time interference indicator (dt), the latter introduced to identify potential disturbances along the interface region. While both variables were examined, the primary objective of the tomographic evaluation was to determine whether the dt indicator could reveal potential interface delamination.

5.2.1 – Representation of the Field Velocity Reconstruction

Through the tomographic reconstruction process, measurements from multiple SA pairs are distributed across a grid representing the beam cross-section in the form of field velocity. As a consequence of this reconstruction, the response measured along individual sensor paths is spatially redistributed and combined with neighbouring measurements, which inevitably reduces spatial accuracy. Localised disturbances detected along a specific ray path are therefore spread across a larger region of the reconstruction grid, causing damage signatures to appear more diffuse in the tomographic images. Despite this limitation, the field velocity plots still provide useful qualitative information regarding the general location of structural disturbances, as regions experiencing cracking are typically represented by zones of reduced velocity within the reconstructed field. However, because the measurements are redistributed across the grid, the precise crack location and the exact load step at which the disturbance first occurred become less clearly defined in the tomographic representation.

Figure 150, Figure 151 and Figure 152 illustrate the field velocity distributions at load stages with separate inaccuracies, which will be discussed.

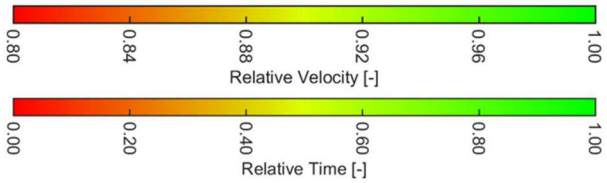
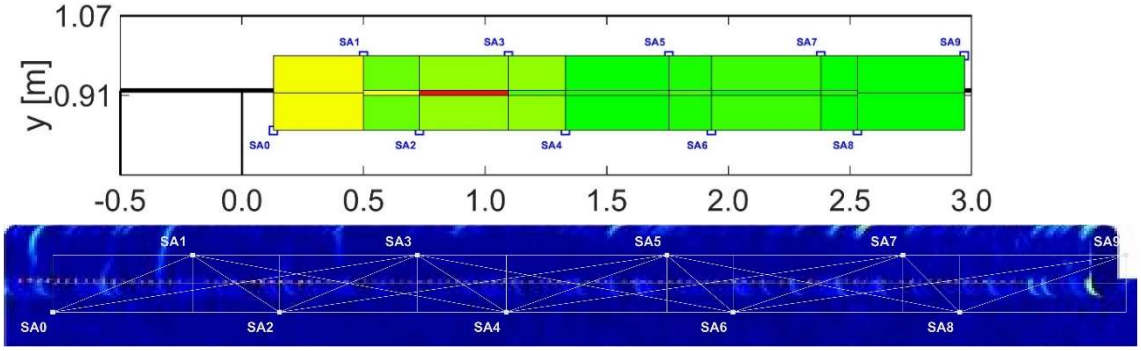
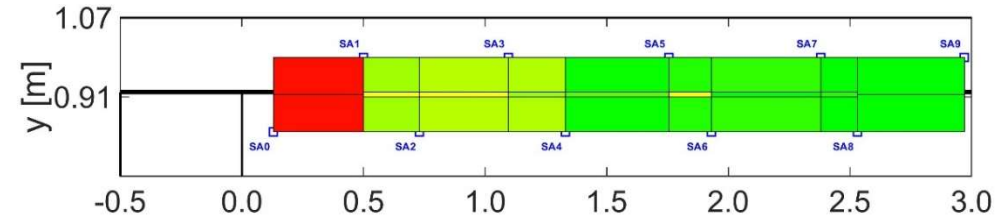


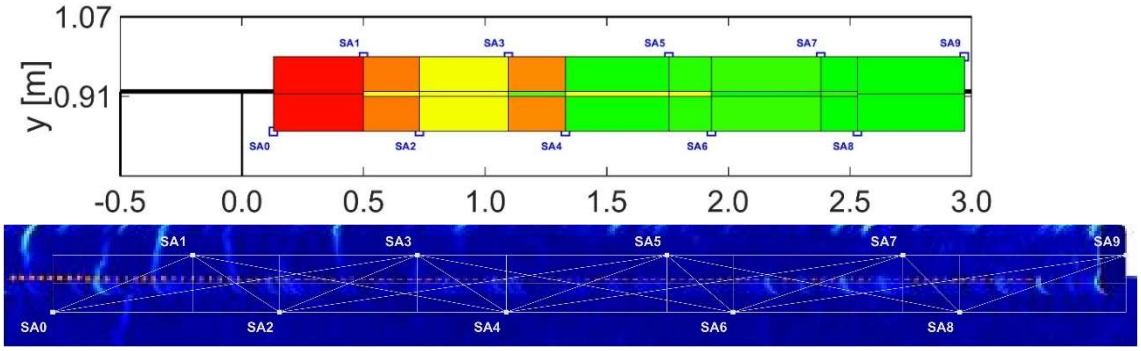
Figure 150, Onset load stages (450-550 kN) tomography Graph of Beam S10H2D, (a)- colour bars of RV&RT-value, (b)-(d) tomography v DIC observations



(b)- 450[kN]



(no DIC Figure)
(c)- 500[kN]



(d)- 550[kN]

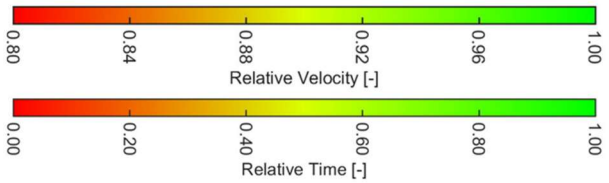
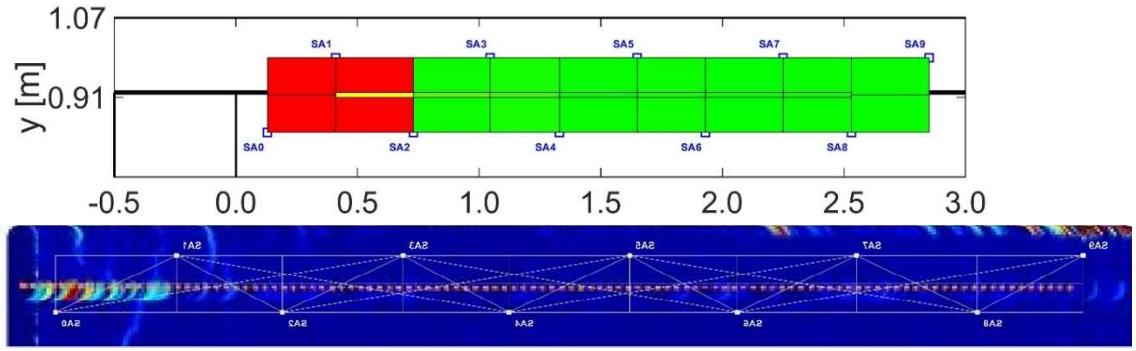
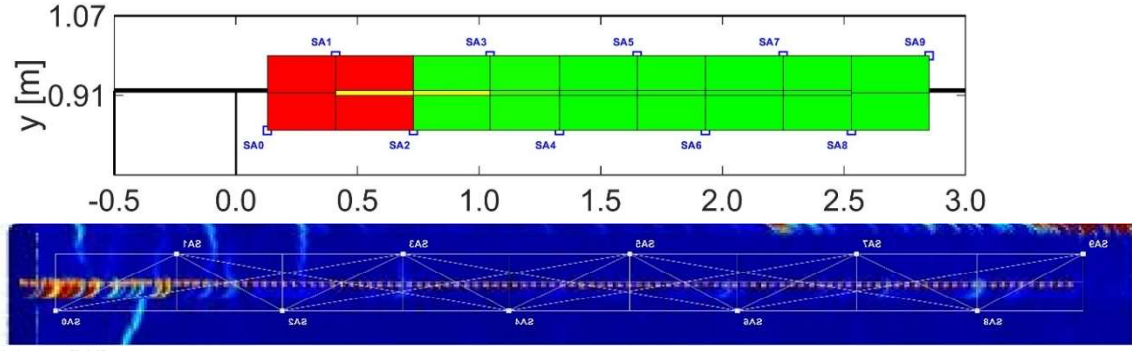


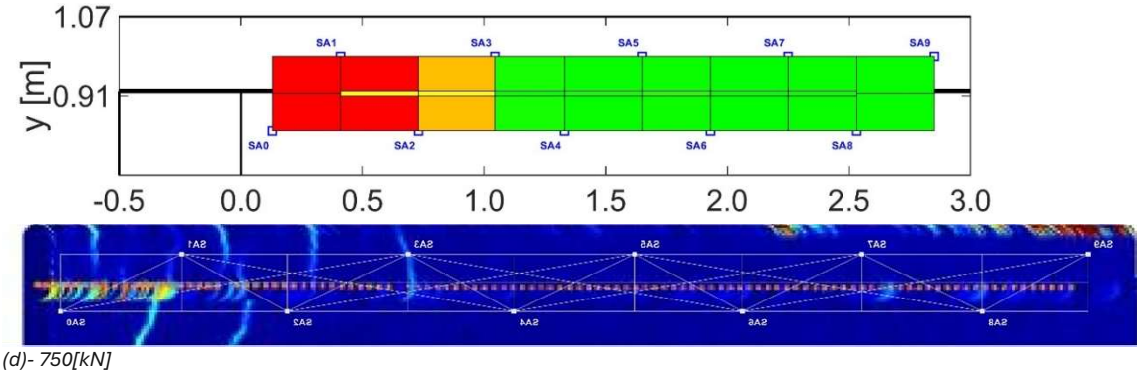
Figure 151, Onset load stages (650-750 kN) tomography Graph of Beam S10H2A, (a)- colour bars of RV&RT-value, (b)-(d) tomography v DIC observations (Flipped over y-axis)



(b)- 650[kN]



(c)- 700[kN]



(d)- 750[kN]

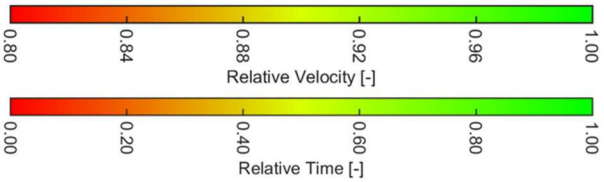
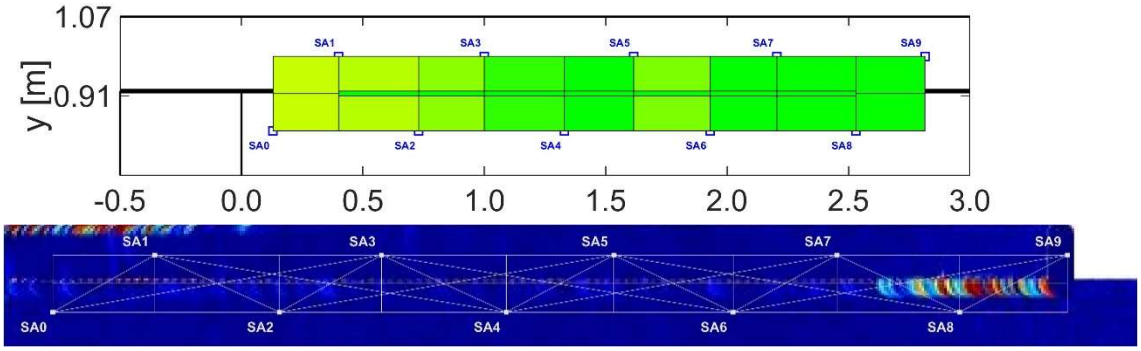
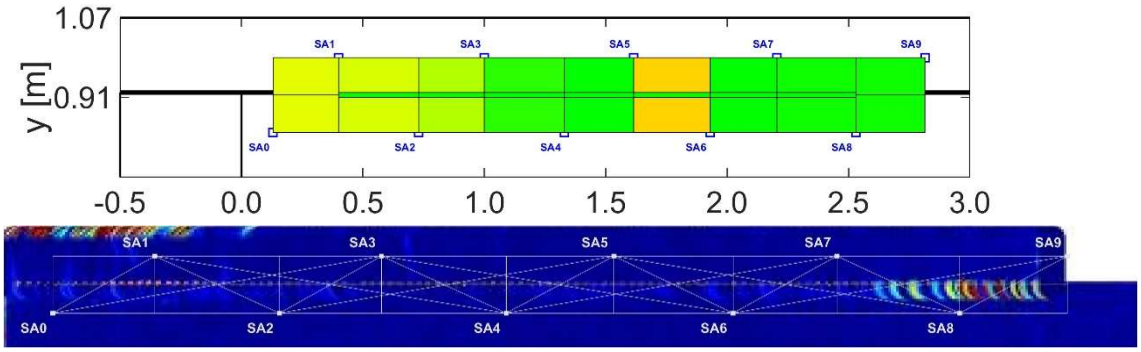


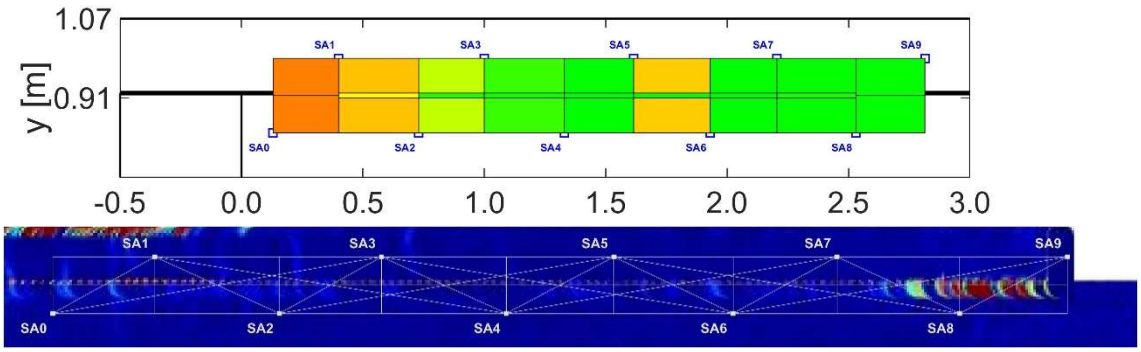
Figure 152, Onset load stages (400-500 kN) tomography Graph of Beam S10H1D, (a)- colour bars of RV&RT-value, (b)-(d) tomography v DIC observations



(b)- 400[kN]



(c)- 450[kN]



(d)- 500[kN]

A further limitation of the tomographic representation is that a consistent threshold or colour scale cannot be clearly defined to represent structural damage. Due to the redistribution of measurements during the reconstruction process, the magnitude of the field velocity reduction does not scale consistently with the severity of the structural disturbance. This behaviour is evident in beam S10H2D (Figure 150), where comparable field velocity reductions are observed for different stages of crack development, despite clear differences in the extent of damage. Additionally, in some cases, a single crack produces a velocity reduction comparable to that caused by multiple cracks or more extensive damage. As a result, the field velocity values cannot be interpreted using a universal threshold.

Figure 151c for beam S10H2A provides an example where a visible crack observed in the DIC measurements is not reflected in the tomographic field velocity distribution, indicating that certain disturbances may be smoothed or partially suppressed during the reconstruction process.
As discussed in Chapter 4, the field velocity results are also influenced by the pairing of the reconstruction fields. Due to the reconstruction method, the top and bottom fields within each pair always share identical velocity values. This behaviour originates from an error in the reconstruction calculation and can introduce interpretation error. In Figure 152d, fields XI and XII (between SA5 & SA6) a reduction in velocity is observed even though no cracking is present in this region in the corresponding DIC measurements, which represents a separate calculation error rather than a physical structural response.

5.2.2 – Representation of the Interface Time Interference Indicator

The interface time interference indicator (dt) is obtained through tomographic processing and therefore represents the most distinctive variable introduced by the tomography approach. Both field velocity and dt are visualised within the same tomographic plots; however, the dt variable provides additional information by highlighting variations in arrival time that may indicate disturbances along the interface region.

Among the three tested beams, S10H2D provided the clearest results. In this beam, the tomographic reconstruction produced well-defined anomalies along the interface region that corresponded with cracking observed in the DIC measurements, as shown in Figure 150. For the other two beams, however, the tomographic results were less consistent. The reconstructed dt patterns were more diffuse, and the indicator often responded at a later stage relative to the onset of delamination, making it difficult to confidently relate the observed variations to interface disturbances.
At present, no definitive threshold value for the dt indicator can be established. While the results remain inconclusive, a relative time value exceeding 2.0 appears to be indicative of interface disturbance or delamination. However, several limitations prevent the definition of an exact threshold. Consequently, a value of dt > 2.0 was adopted for visualisation purposes in the tomographic plots, while acknowledging the associated uncertainty.

Limitations observed in the Interface Time Interference Indicator

A first limitation, as discussed in Chapter 4, is the observed decrease in dt values at later load stages despite continued crack development. This behaviour is physically inconsistent and suggests a reduction in the reliability of the tomographic reconstruction under more advanced damage conditions. In addition, dt values may return towards values close to 1.0 or exhibit non-monotonic fluctuations between load steps, further complicating interpretation.
A second issue concerns the inconsistent spatial distribution of dt anomalies. In several instances, individual fields (e.g. dt1 and dt3) exhibit elevated values, while adjacent fields (e.g. dt2) do not, despite DIC observations indicating continuous delamination across all corresponding regions (Figure 150). This fragmented response reduces the interpretability of the results and suggests that the observed behaviour may, at least in part, originate from inaccuracies in the tomographic calculation rather than actual structural variation. This effect was observed to occur more frequently at later load stages, as discussed in Sub-chapter 4.3, where increasing structural deformation and compounding errors likely amplified reconstruction-related inaccuracies.

Reconstruction Limitations in the Interface Time Interference Indicator

A third limitation relates to the formulation of the tomographic system of equations (see Chapter 2.3.3.3). Within this formulation, dt is treated as an additional unknown variable, meaning it can be adjusted to satisfy the system of equations. During the development of the method, it was observed that solutions could be obtained with non-physical values ($dt < 0$), necessitating the introduction of an additional boundary condition ($dt > 0$). This indicates that the solution space is not inherently constrained to physically meaningful results, and that dt values may partially reflect numerical artefacts introduced during the solution process.
Finally, the geometry of the sensor configuration imposes an additional limitation on the tomographic reconstruction. Although the NRMSD values for beams S10H2A and S10H1D may indicate a satisfactory fit of the solution, the corresponding tomographic results remain inaccurate at higher load levels. This suggests that a low NRMSD does not necessarily guarantee physically reliable results, particularly when the underlying system of equations becomes increasingly sensitive to complex cracking patterns and limited measurement coverage.

Overall, while the dt indicator demonstrates potential for identifying interface disturbances, its interpretation remains constrained by limitations in the tomographic reconstruction process. Consequently, dt anomalies should be interpreted with caution and always in combination with complementary observations, such as DIC measurements and Ray Path analysis.

5.3 – Combined Interpretation of the Ray-path Graphs

This sub-chapter evaluates the combined use of CWI and UPV for damage detection and localisation. As outlined in Chapter 1, a central goal of this thesis is to assess how CWI-based indicators can be used for early-stage damage detection, while UPV-based indicators provide subsequent localisation of developed cracking. In Chapter 4 and earlier sections of Chapter 5, the ε, CC, and RV graphs were analysed individually to assess their respective sensitivity and behaviour. In this sub-chapter, these indicators are examined in a combined framework to evaluate how their complementary characteristics can be leveraged for improved interpretation of crack initiation and propagation.
The combined examination of CWI and UPV ray-path graphs is discussed primarily with reference to beam S10H2D, as this specimen most clearly demonstrates the effectiveness of the approach in terms of early detection and localisation. Beam S10H1D is additionally considered to highlight the limitations and inconsistencies that can arise in the combined interpretation. Beam S10H2A is not discussed separately, as it exhibits comparable behaviour and trends to S10H2D, offering no additional insight into the performance of the combined approach.

In Figure 153, Figure 154 and Figure 155, beam S10H2D is shown at load steps of 350, 400, and 450 kN, corresponding to the onset of cracking. For each load step, DIC, CC and RV ray-path graphs are presented. The ε graphs are not considered in this analysis, as shown in Chapter 4, where they exhibit high sensitivity and respond significantly prior to crack formation, limiting their usefulness for localisation. The cracks discussed are highlighted in the corresponding DIC figures.

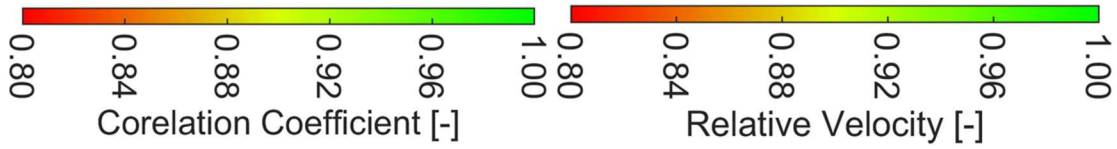
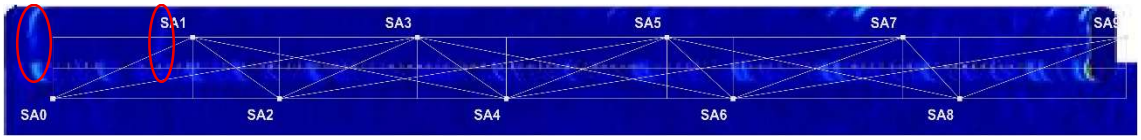
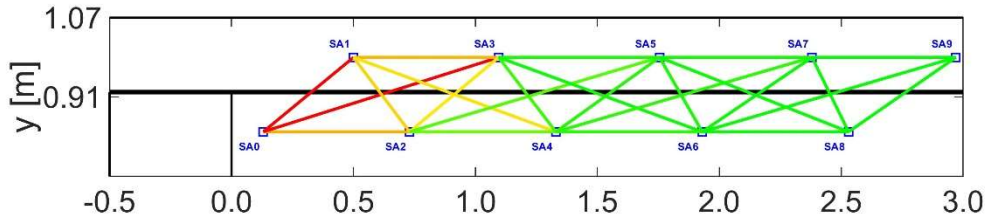


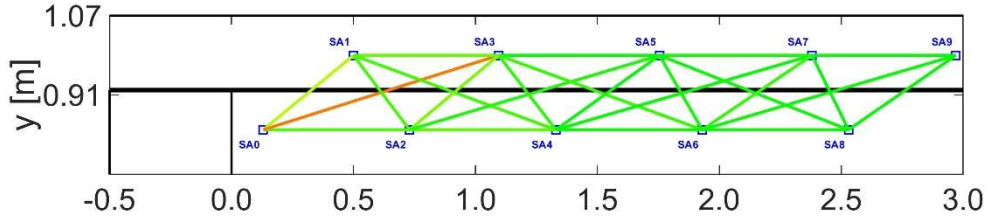
Figure 153, Combined Graphs v DIC observations of beam S10H2D at load 350 [kN], (a)- colour bars CC-Graph and Relative Velocity Graph, (b)- DIC observation, (c)- CC-Graph, (d)- RV-Graphs



(b) – DIC observation at 350[kN]



(c)- CC-Graph at 350[kN]



(d)- RV-Graph at 350[kN]

At a load of 350 kN, marking the onset of cracking, two flexural cracks are observed to the left of SA1. The CC graphs provide the first meaningful indication of structural change, as reflected by a shift in the colour scale along specific ray paths, indicating decorrelation of the waveform. From these CC graphs, a gradient in the spatial distribution of change can be observed, suggesting the approximate location of crack initiation. The RV graphs demonstrate a higher degree of localisation, with SA pair 0–1, the closest to the observed cracks, indicating a clear change. However, SA pair 0–3 shows a relatively stronger response despite no visible cracking or delamination along this path, with similar behaviour observed for pair 1–3.

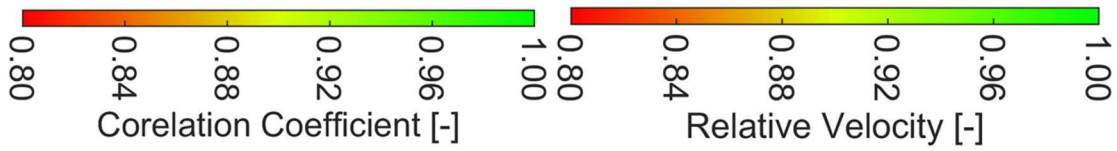
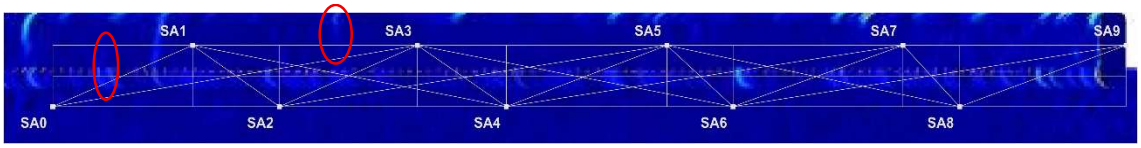
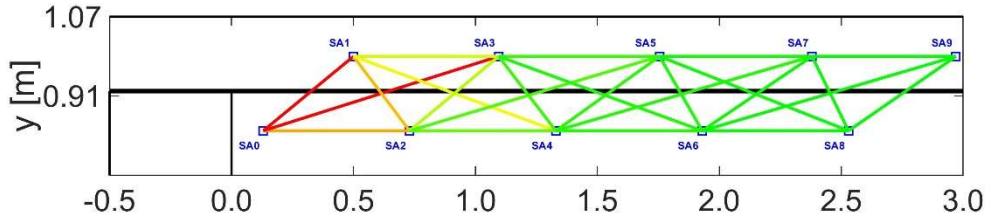


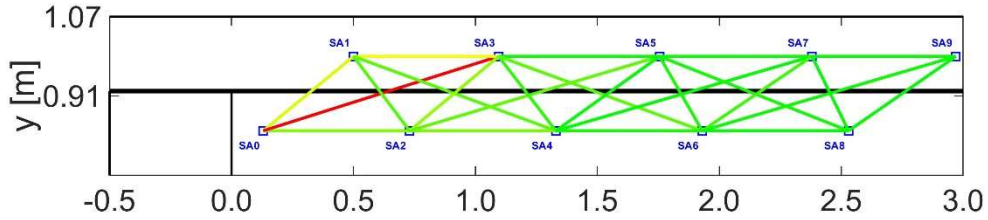
Figure 154, Combined Graphs v DIC observations of beam S10H2D at load 400 [kN], (a)- colour bars CC-Graph and Relative Velocity Graph, (b)- DIC observation, (c)- CC-Graph, (d)- RV-Graphs



(b) – DIC observation at 400[kN]



(c)- CC-Graph at 400[kN]



(d)- RV-Graph at 400[kN]

At 400 kN, a new flexural crack develops between SA0 and SA1, while the previously identified cracks continue to propagate. This progression is reflected in the CC graphs through an increased magnitude and spatial spread of decorrelation along the affected ray paths. The RV graphs again provide localisation of the three flexural cracks between SA0 and SA1, with corresponding SA pairs indicating increased response. Additionally, a flexural crack appears to the left of SA3; however, this crack is only weakly expressed in the CC graphs at this load step. A slight response in the corresponding RV ray paths may be associated with this newly formed crack.

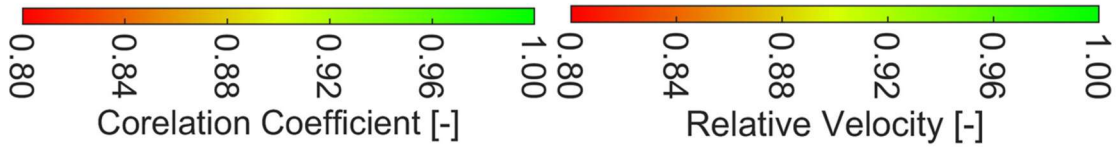
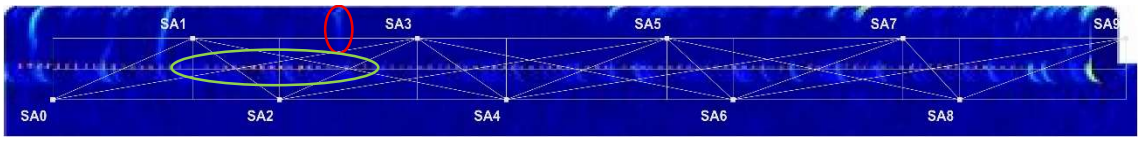
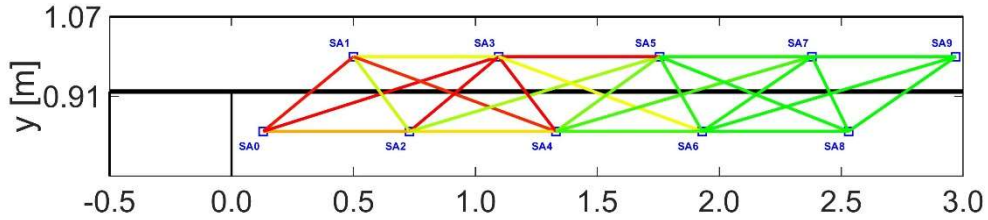


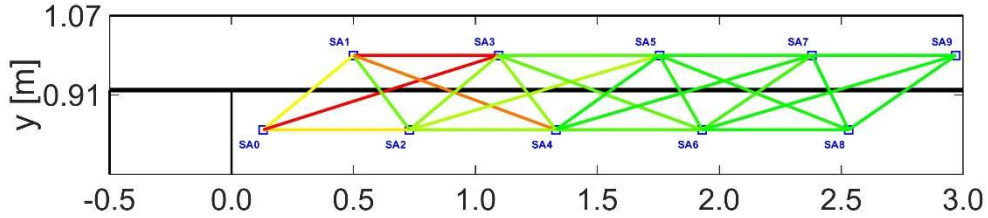
Figure 155, Combined Graphs v DIC observations of beam S10H2D at load 450 [kN], (a)- colour bars CC-Graph and Relative Velocity Graph, (b)- DIC observation, (c)- CC-Graph, (d)- RV-Graphs



(b) – DIC observation at 450[kN]



(c)- CC-Graph at 450[kN]



(d)- RV-Graph at 450[kN]

At 450 kN, the existing cracks further propagate and widen. In the CC graphs, additional SA pairs, particularly 0–2 and multiple diagonal pairs between SA0 and SA4, exhibit increased decorrelation, indicating a broader distribution of internal changes. Although no shear crack is yet observed in the DIC, the RV graph similarly shows increased response in SA pair 0–2. Furthermore, both DIC and RV graphs indicate the development of delamination near the intersection of SA pairs 0–3 and 1–4. The flexural crack located to the left of SA3 extends towards SA pair 1–3, which is also reflected in the RV graph. In contrast, the flexural cracks between SA0 and SA1 continue to produce a comparatively weaker response in both CC and RV graphs.

For load levels between 350 and 450 kN, the results demonstrate that CC graphs provide effective early detection, as they are the first to indicate changes through progressive decorrelation reflected in the colour scale. RV graphs offer improved localisation, particularly for flexural cracks, by identifying specific SA pairs associated with damage. Overall, the combined use of CC and RV graphs confirms that early-stage damage can be detected prior to full crack development, with localisation improving as cracks propagate.

In Figure 156, Figure 157 and Figure 158, the DIC, CC, and relative velocity ray-path graphs corresponding to load steps of 500, 550, and 600 kN are presented, with the discussed cracks highlighted in the DIC figures.

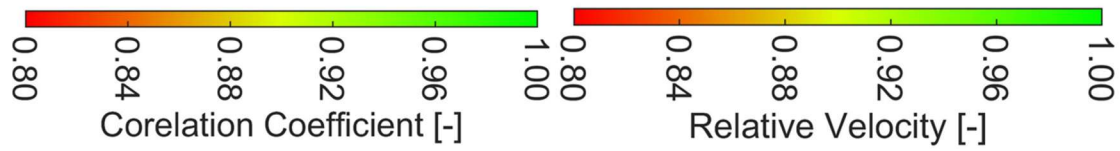
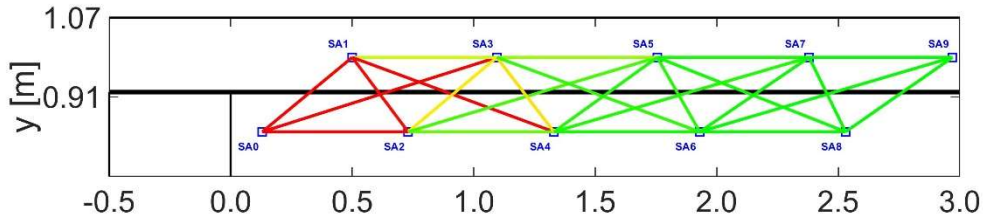
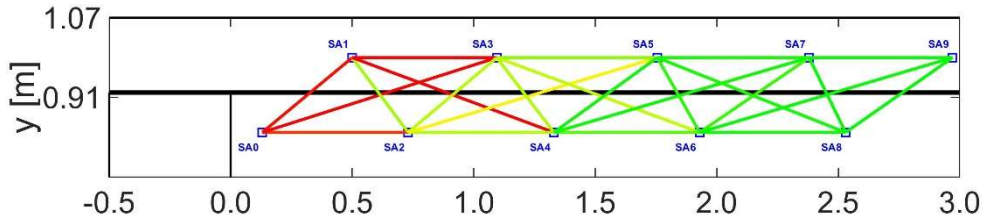


Figure 156, Combined Graphs v DIC observations of beam S10H2D at load 500 [kN], (a)- colour bars CC-Graph and Relative Velocity Graph, (b)- DIC observation, (c)- CC-Graph, (d)- RV-Graphs

(b)- (No DIC Figure)



(c)- CC-Graph at 500[kN]



(d)- RV-Graph at 500[kN]

At 500 kN, no DIC image is available; however, a pronounced response is observed in both CC and RV graphs for SA pair 0–2, where a strong increase in magnitude (red region) is evident. This suggests significant internal change along this path and possible occurrence of shear cracking, which will be further examined at the subsequent load step of 550 kN.

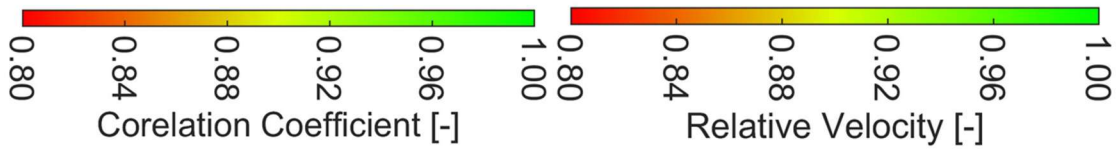
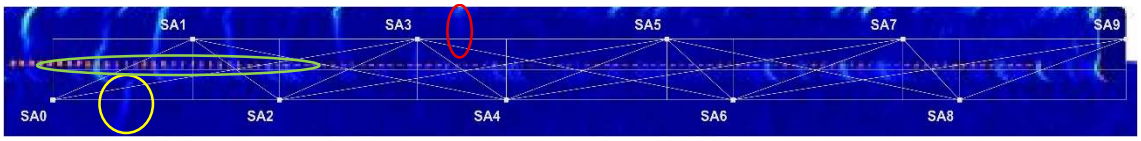
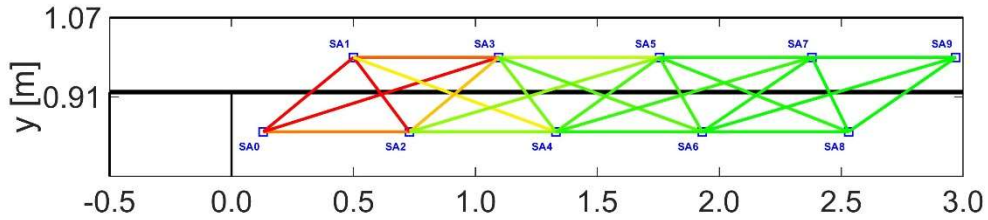


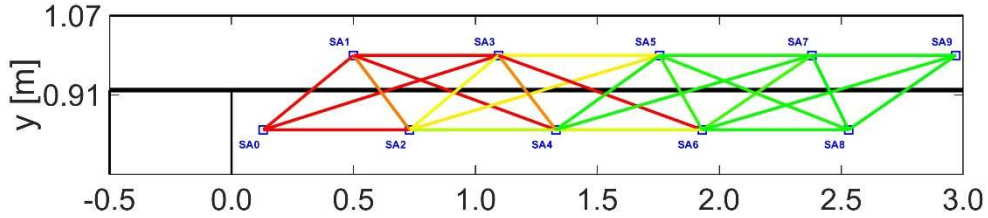
Figure 157, Combined Graphs v DIC observations of beam S10H2D at load 550 [kN], (a)- colour bars CC-Graph and Relative Velocity Graph, (b)- DIC observation, (c)- CC-Graph, (d)- RV-Graphs



(b) – DIC observation at 550[kN]



(c)- CC-Graph at 550[kN]



(d)- RV-Graph at 550[kN]

At 550 kN, a clear shear crack becomes visible to the right of SA0 in the DIC observations. Additionally, indications of a new flexural crack to the right of SA3 can be inferred from both CC and RV graphs. However, the SA pairs located closest to this crack do not exhibit a proportionally strong response in either graph, limiting localisation accuracy. At the same time, the diagonal SA pairs, which previously indicated change at 500 kN, continue to show elevated responses in both CC and RV graphs. These responses may be attributed to the development or expansion of interface delamination; however, the contribution of worsening flexural cracks in this region cannot be excluded, making interpretation non-unique.

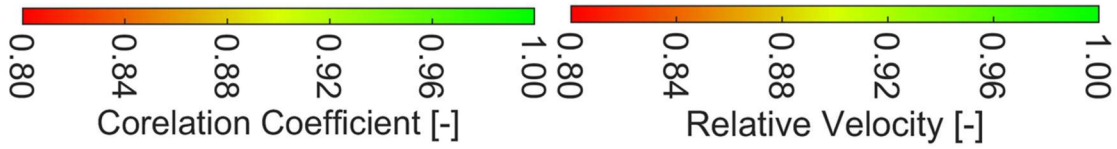
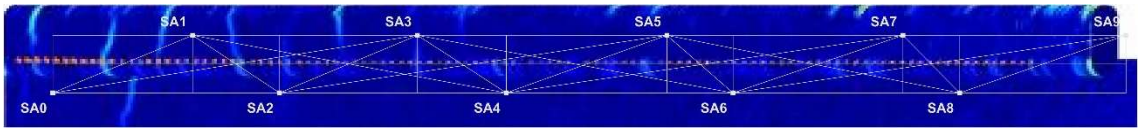
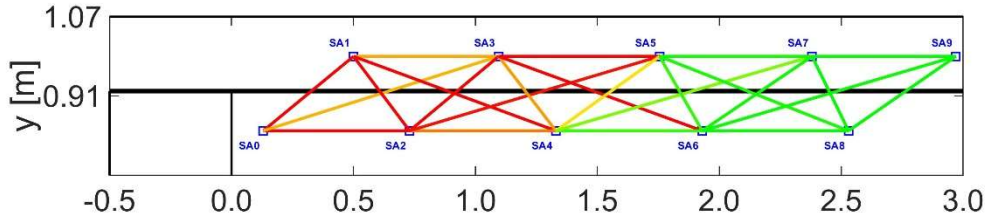


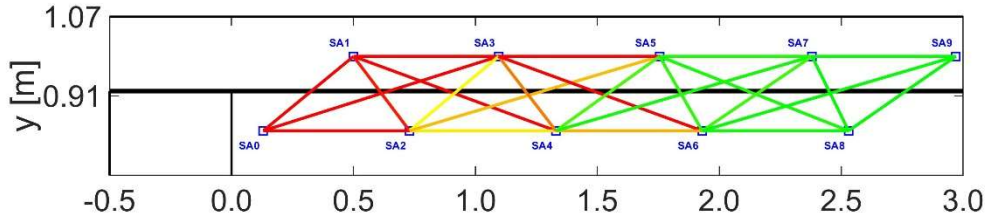
Figure 158, Combined Graphs v DIC observations of beam S10H2D at load 600 [kN], (a)- colour bars CC-Graph and Relative Velocity Graph, (b)- DIC observation, (c)- CC-Graph, (d)- RV-Graphs



(b) – DIC observation at 600[kN]



(c)- CC-Graph at 600[kN]



(d)- RV-Graph at 600[kN]

At 600 kN, all SA pairs between SA0 and SA5 exhibit a strong response in both CC and RV graphs, indicating widespread internal change. The DIC observations confirm that no new cracks form at this stage; instead, existing cracks continue to propagate and widen, and delamination further extends. Due to the high level of structural damage, the ultrasonic response becomes saturated, and the ability to localise or isolate individual cracks is significantly reduced, as both CC and RV graphs indicate extensive and overlapping regions of change.

For load levels between 500 and 600 kN, both CC and RV graphs exhibit strong and widespread responses, reflecting significant structural degradation. While early indications of damage at 500 kN correspond to later observed shear cracking and possible delamination at 550 kN, the interpretation becomes increasingly ambiguous due to overlapping damage mechanisms. At higher load levels, particularly at 600 kN, signal saturation occurs, and both CC and RV graphs lose their ability to distinguish between individual damage sources. As a result, localisation capability diminishes, and the indicators primarily reflect the overall extent of structural degradation rather than discrete crack identification.

Figures XX, XX, and XX present the DIC, CC, and relative velocity ray-path graphs for beam S10H1D at load steps of 400 and 450 kN, with the discussed cracks highlighted in the corresponding DIC images.

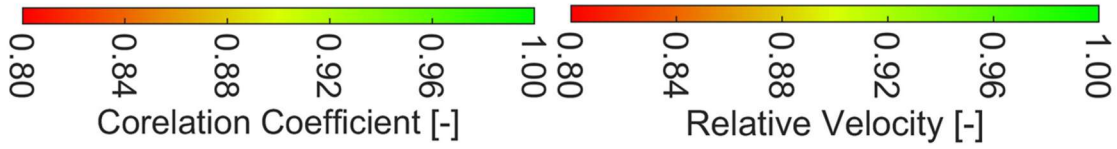
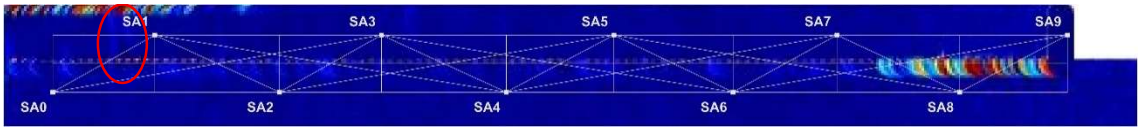
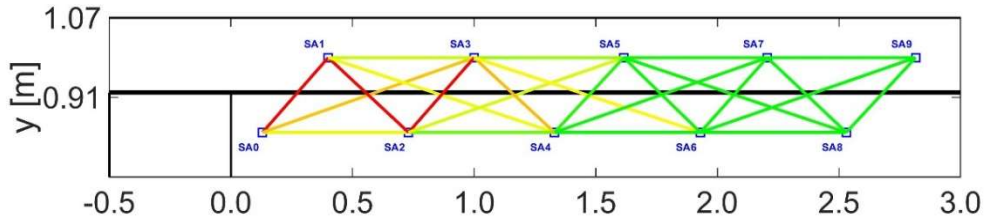


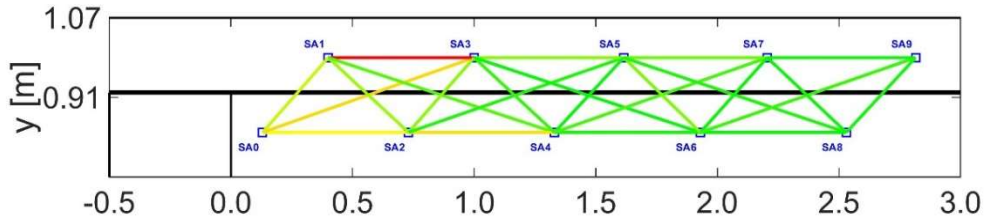
Figure 159, Combined Graphs v DIC observations of beam S10H1D at load 400 [kN], (a)- colour bars CC-Graph and Relative Velocity Graph, (b)- DIC observation, (c)- CC-Graph, (d)- RV-Graphs



(b) – DIC observation at 400[kN]



(c)- CC-Graph at 400[kN]



(d)- RV-Graph at 400[kN]

At 400 kN, the DIC observations show two initial and relatively faint damage features: a flexural crack located to the left of SA1 and early-stage delamination between SA0 and SA1. The CC graph indicates clear decorrelation in SA pairs 0–1, 1–2, and 2–3. The response in pair 0–1 may be associated with the flexural crack, while the diagonal pairs (1–2 and 2–3) likely correspond to the developing delamination. However, the RV graph suggests a different damage pattern, indicating a pronounced change between SA1 and SA3, where no visible crack is present in the DIC. Additionally, the lower SA pairs 0–2 and 2–4 show notable responses, suggesting damage in regions where no cracking or delamination is observed. This indicates a reduced correspondence between RV responses and physically observed damage at this stage.

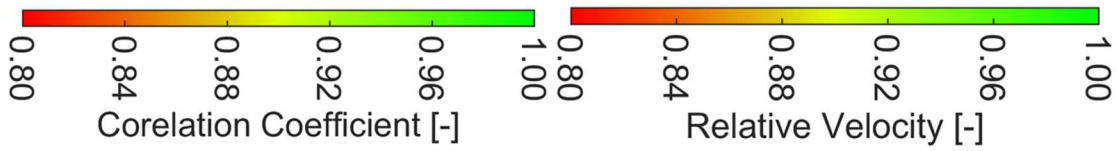
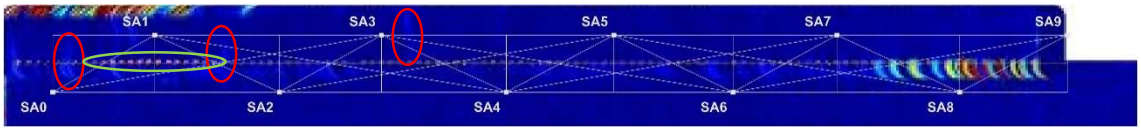
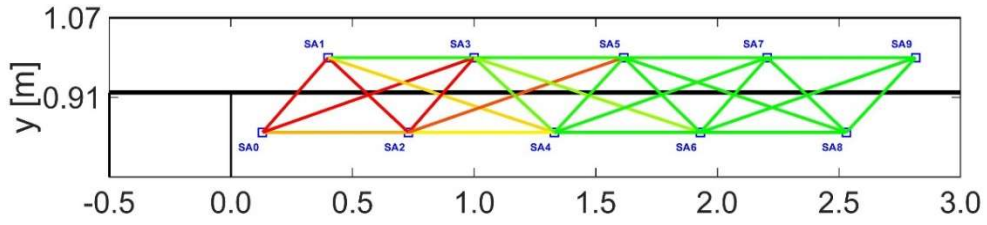


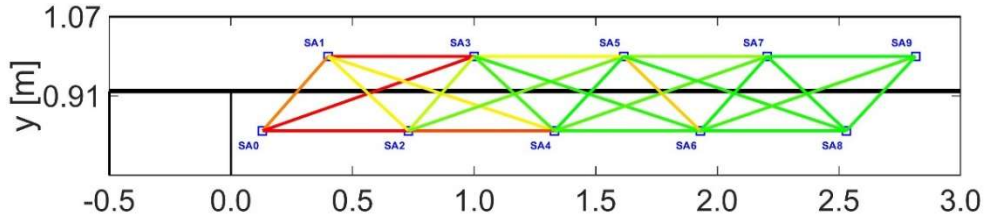
Figure 160, Combined Graphs v DIC observations of beam S10H1D at load 450 [kN], (a)- colour bars CC-Graph and Relative Velocity Graph, (b)- DIC observation, (c)- CC-Graph, (d)- RV-Graphs



(b) – DIC observation at 450[kN]



(c)- CC-Graph at 450[kN]



(d)- RV-Graph at 450[kN]

At 450 kN, three additional flexural cracks develop, and the delamination between SA0 and SA2 extends. The CC graph captures the progression of damage between SA0 and SA2 and identifies the flexural cracks located to the left and right of SA1. However, it does not clearly detect the newly formed flexural crack to the right of SA3. The RV graph reflects the presence of flexural cracking and the extended delamination, with strong responses observed in SA pairs 0–3 and 1–3 (red) and a moderate response in 0–1 (orange). Despite this, the previously noted inconsistencies in the lower SA pairs (0–2 and 2–4) become more pronounced, as these pairs now exhibit strong responses (red) without corresponding physical damage in the DIC observations.

Compared to beam S10H2D, beam S10H1D demonstrates a lower level of agreement between ultrasonic indicators and observed damage. While the CC graph provides a reasonably consistent indication of crack development and delamination, the RV graph exhibits a higher number of false-positive responses and inconsistencies, particularly in the lower SA pairs. These discrepancies reduce the reliability of localisation and suggest that, for this beam, the RV-based interpretation is more sensitive to non-damage-related effects or complex wave interactions, leading to a less accurate representation of actual structural behaviour.

5.4 – Key Observations & Discussion Points on the Ultrasonic Indicators

Ray Path-based indicators and representation

- It appears that ε and CC provide distinct information for early detection: ε is sensitive to early-stage disturbances, possibly related to microcracking or stress redistribution prior to visible cracking, but lacks specificity for crack initiation, whereas CC provides a more consistent response to crack onset. CC is best interpreted through relative reductions rather than a fixed threshold value, while the ε threshold remains applicable but less selective.
- The Ray Path graphs do not clearly represent the distributed nature of the CWI response along the coda wave propagation paths; however, ε still reflects gradual variations associated with stress redistribution over the load stages, while CC exhibits a spatial gradient, transitioning from lower values (red) near disturbance locations to higher values (green) in unaffected regions.
- UPV-based relative velocity corresponds well in detecting flexural and shear crack development for the top and bottom SA pairs, for which a threshold value of 0.80 is effective in identifying crack-related disturbances.
- Diagonal SA pairs provide less reliable identification of crack types due to their orientation and the increased likelihood of intersecting multiple crack types; this configuration is therefore not optimised for detecting interface-related damage.
- Diagonal SA pairs provide less reliable identification of crack types due to their orientation and the increased likelihood of intersecting multiple crack types; this configuration is therefore not optimised for detecting interface-related damage.
- The angle at which the ray path intersects the interface influences the response: lower intersection angles are more affected by distributed cracking along the path, while steeper angles tend to produce clearer and more localised indications when the interface is involved.
- The position of the SA groups along the beam length influences interpretation, as regions with higher crack density led to more complex responses, reducing the ability to distinguish individual crack mechanisms.

Tomographic indicators and representation

- The tomographic reconstruction process reduces spatial accuracy, and the resulting field velocity does not consistently capture the onset of damage nor scale with its severity, preventing the definition of a universal threshold; consequently, both the type and magnitude of damage are difficult to interpret from the reconstructed fields.
- The dt indicator shows potential for identifying interface disturbances, particularly at early load stages; however, a consistent threshold cannot be defined, although values of $dt > 2.0$ appear indicative. Interpretation is further complicated by the role the dt variable assumes in the formulation of the tomographic system, where it functions as a free parameter within the solution process, introducing the possibility of non-physical results.
- The tomographic results are affected by multiple model-related limitations, including the enforced pairing of top and bottom fields, sensitivity to numerical formulation (e.g. dt as an unconstrained variable), and the limited reliability of validation metrics such as NRMSD. These factors collectively indicate that the observed results may contain systematic reconstruction errors, particularly at higher load levels.

Combined indicators approach

- The combined interpretation of CWI-based indicator CC with UPV-derived relative velocity proves effective overall, particularly at lower load levels where crack density is limited. In these early stages, changes in the indicators can be more confidently attributed to individual cracking events.
- This strength diminishes as damage progresses and cracking becomes more distributed, but the approach remains sufficiently robust in beams such as S10H2D, where consistent trends between indicators allow for a coherent interpretation of crack initiation and localisation.
- At higher load levels, the combined interpretation remains valuable for identifying regions further along the beam where cracking has not yet initiated. Subtle changes in CWI indicators, when supported by emerging trends in UPV along adjacent transmission paths.
- With respect to interface delamination, the combined approach provides indicative—but not definitive—identification. Consistent responses across CWI and UPV along interface-crossing paths can suggest degradation at the interface; however, distinguishing delamination from nearby flexural or shear cracking remains challenging. Reliable identification therefore depends on convergence between multiple indicators and careful consideration of path orientation, rather than reliance on any single measure.
- However, the method is not without limitations, as demonstrated by S10H1D. Here, discrepancies between indicators—most notably in the UPV response—introduce ambiguity, with certain transmission paths indicating damage not supported by CWI or DIC. This highlights that, while the combined approach enhances reliability, it does not fully eliminate misinterpretation, particularly in more complex cracking scenarios or less favourable sensor configurations.

Key Limitations in the Methodology

A limitation of the experimental dataset is that only three out of the seven tested beams provided data of sufficient quality for detailed analysis. Multiple datasets were affected by signal noise, of which two beams exhibited noise levels so severe that the ultrasonic response was effectively obscured, rendering the data unusable for both UPV and CWI analyses.
In two additional beams, the signal quality was insufficient for consistent application of the Hinkley Criterion–based picker. This method is used to identify the onset of the ultrasonic waveform for UPV and to determine the centre of the coda window for CWI analysis. In these cases, the algorithm frequently failed to identify the correct reference points, requiring manual intervention. While the data remained usable, the need for repeated manual selection significantly reduced processing efficiency and limited the practicality of the automated workflow.

A key limitation of the monitoring approach is that, while flexural and shear cracking could be detected consistently using UPV-based indicators and ray-path representations, the detection of interface delamination proved significantly more challenging. CWI-based indicators were sensitive to anomalies in the interface region; however, these responses could not be uniquely attributed to delamination without reference to DIC observations, as localisation requires path-based interpretation through UPV. Particularly for diagonal sensor pairs crossing the interface, was unable to isolate interface-specific damage due to the interaction of multiple crack types along a single propagation path and the unfavourable orientation of interface-related damage relative to wave propagation

If the diagonal SA pairs are examined further, a distinction can be made based on the inclination of the ray path relative to the interface, separating sharp (SA pairs: 1–2, 2–3, 3–4, 4–5) and shallow (SA pairs: 0–3, 1–4, 2–5) pairs. Sharp pairs intersect the interface at steeper angles, while shallow pairs cross at lower angles. This geometric relationship is critical, as the sensitivity of ultrasonic velocity to cracking depends on the angle of intersection: ray paths that intersect damage closer to perpendicular experience stronger scattering and thus larger reductions in relative velocity.
When considering only cases where clear signs of delamination are present, the sharp SA pairs demonstrate more reliable detection behaviour. In multiple instances, sharp pairs indicate delamination with consistent reductions in relative velocity (RV < 0.90), although not always exactly at the onset load; rather, once this threshold is reached, delamination can be identified with a degree of confidence. This relatively consistent threshold provides a clear and interpretable indicator of delamination. In contrast, shallow pairs may show earlier deviations (often around RV ≈ 0.95), but such minor reductions are not sufficiently definitive to confirm delamination and may occur without clear supporting evidence.

Although interface delamination could not be directly isolated, its presence may be inferred indirectly through the structural response observed in both the ultrasonic data and DIC measurements. A concentration of flexural and shear cracks in the vicinity of the interface indicates a deterioration of composite action, which is commonly associated with reduced interface shear transfer capacity and the onset of debonding or delamination. In this context, interface damage may be interpreted not as a single identifiable feature, but as a condition reflected by the combined presence and interaction of multiple crack types. This behaviour can also be qualitatively observed in the ultrasonic graphs, where clustered disturbances or gradients in indicator response near the interface region suggest a loss of structural continuity.

Recommendations for future work

To address the identified limitations, several improvements are recommended for future research. Signal noise remains a critical challenge, particularly for automated processing; therefore, improved sensor coupling and signal conditioning techniques should be explored, alongside more robust picking algorithms capable of handling low signal-to-noise ratios.
With respect to sensor configuration, the current diagonal paths proved insufficient for isolating interface-specific behaviour. Future studies should consider arranging Smart Aggregate pairs such that wave propagation paths intersect the interface at near-perpendicular angles. This configuration would increase sensitivity to interface discontinuities and reduce interference from flexural and shear cracks developing along the beam length.
Furthermore, refinement of the tomographic reconstruction framework is required, particularly in addressing numerical stability and reducing sensitivity to structural deformation. Incorporating constraints or adaptive geometry updates within the reconstruction process may improve the reliability of tomographic indicators such as *dt*, especially at higher load stages.

Chapter 6 – Conclusions

Smart Aggregate–based monitoring, combined with ultrasonic processing techniques, shows strong potential as a structural health monitoring approach for concrete structures. This study has investigated the applicability of these techniques for assessing cracking behaviour in composite precast girder bridges made continuous, with particular focus on the region at and around the concrete-to-concrete interface between the precast inverted T-beam and the cast in-situ top layer.

The results demonstrate that ultrasonic measurements can capture changes in structural behaviour across different stages of damage development, from the onset of internal disturbances to the formation of larger, visible cracks. However, not all damage mechanisms were observed with equal clarity: flexural and shear cracking were identified with relatively high consistency, whereas isolating interface delamination proved significantly more challenging. While Coda Wave Interferometry (CWI) based indicators showed high sensitivity to early changes and Ultrasonic Pulse Velocity (UPV) provided a clear localised response to developed cracking, this study observed that distinguishing interface-specific damage from other structural responses remained difficult.

Consequently, although Smart Aggregate–based ultrasonic monitoring presents a valuable tool for detecting and interpreting structural changes, its current application is best suited for qualitative assessment and trend analysis rather than precise quantitative evaluation. The integration of multiple indicators and visualisation techniques improves interpretability, but further refinement is required before the method can function as a fully reliable standalone monitoring system for complex structural behaviour.

The following sections present a comprehensive evaluation of the findings in relation to the research questions.

What information can be obtained from Smart Aggregate–based ultrasonic measurements regarding crack and interface behaviour in continuous composite precast girder bridges made continuous, and how do the resulting ultrasonic indicators provide distinct yet complementary insight on actual crack and interface behaviour of the structure?

- Smart Aggregates can be directly embedded at critical structural regions, such as the concrete-to-concrete interface, enabling in-situ ultrasonic monitoring of internal structural behaviour.
- UPV evaluates early wave arrivals to determine velocity changes associated with stiffness reduction and crack formation, while CWI (via CC and ε) captures distributed changes linked to material heterogeneity, stress redistribution, and microcrack development.
- Combined, CWI provides early sensitivity to subtle, distributed damage, whereas UPV enables localisation and confirmation of developed cracks, offering complementary insight into crack evolution and interface behaviour.

How can Smart Aggregate–based ultrasonic measurements be processed and visualized in a clear and meaningful manner to highlight the relevant ultrasonic indicators and support their interpretation with respect to crack and interface behaviour?

- Ray-path representation maps ultrasonic indicators along SA transmission paths, preserving the physical wave–structure interaction, where relative velocity clearly identifies crack presence through path-specific changes.
- Although CWI reflects distributed changes rather than localised effects, it can be interpreted in ray-path form, where CC highlights disturbance zones or gradients and ε visualises gradual stress redistribution across load stages.

- Tomographic reconstruction converts path-based ultrasonic data into spatially continuous fields, enabling visualisation of damage distribution, while spreading individual ray-path responses over the domain, causing damage effects to appear across neighbouring regions rather than at a single physical location.
- Within this framework, the interface time interference (dt) indicator is introduced as an additional variable in the tomographic system of equations, representing cumulative time-delay effects associated with interface-related changes.

To what extent can ultrasonic indicators which are derived from Smart Aggregate measurements be validated against established measurement techniques to substantiate interpretations of cracking and interface delamination?

- The CWI indicator CC shows strong qualitative agreement with crack initiation observed in DIC measurements, with a threshold of CC < 0.8 providing the most consistent and reliable indication of crack onset.
- The ε indicator responds to early disturbances, likely associated with stress redistribution or microcracking, which were not explicitly captured or directly observable in the DIC measurements.
- UPV-derived relative velocity correlates well with developed cracks observed in DIC measurements and enables reliable localisation of flexural and shear cracks when the ray-path orientation aligns with the crack direction; a threshold of relative velocity < 0.8 was found to be most effective.
- However, interface delamination could not be distinctly isolated using UPV, as diagonal SA pairs traversing the interface intersect multiple crack types and are influenced by propagation angle; for steeper diagonal paths, a threshold of relative velocity < 0.90 was found to provide the most consistent indication of delamination.
- Tomographic indicators, including dt, show potential for identifying interface-related behaviour, particularly at early load stages, where a threshold of dt > 2.0 was found to be most indicative; however, interpretation remains limited by reconstruction uncertainties and numerical sensitivities.
- The tomographic reconstruction process introduces increasing inaccuracies at higher load stages, as the method assumes a stable geometry, whereas structural deformation becomes more significant with increasing load.

References

Achenbach, J. D. (1999). *Wave propagation in elastic solids*. Elsevier.

Agdas, D., Asce, M., Rice, J. A., Martinez, J. R., & Lasa, I. R. (2015). *Comparison of Visual Inspection and Structural-Health Monitoring As Bridge Condition Assessment Methods*. https://doi.org/10.1061/(ASCE)CF

Aggelis, D. G., & Shiotani, T. (2007). Repair evaluation of concrete cracks using surface and through-transmission wave measurements. *Cement and Concrete Composites*, *29*(9), 700–711. https://doi.org/10.1016/j.cemconcomp.2007.05.001

Aguero, M., Doyle, D., Mascarenas, D., & Moreu, F. (2023). Visualization of real-time displacement time history superimposed with dynamic experiments using wireless smart sensors and augmented reality. *Earthquake Engineering and Engineering Vibration*, *22*(3), 573–588. https://doi.org/10.1007/s11803-023-2184-x

Alampalli, S., & Ettouney, M. (2008). Role of structural health monitoring in bridge security. *Bridge Structures*, *4*(3–4), 143–154. https://doi.org/10.1080/15732480802399417

Birkeland, P. W., & Birkeland, H. W. (1966). Connections in Precast Concrete Construction. *JOURNAL OF THE AMERICAN CONCRETE INSTITUTE*, 345–368.

Borges, A., Yang, E. A., Roosen, Y. ;, & Hendriks, M. A. (2024). *Concrete-to-concrete interface behaviour in precast girder bridges made continuous deficiencies and challenges*. APA. https://www.openaccess.nl/en/you-share-we-take-care

Burns, N. H. (1966). Development of Continuity Between Precast Prestressed Concrete Beams. *PCI Journal*, *11*(3), 23–36. https://doi.org/10.15554/pcij.06011966.23.36

Calavera, J., De Chefdebien, A., Fernández-Ordóñez, D., Gasperi, A., Ley, J., Mönnig, F., Passeman, P., Quartel, C., Sasek, L., Tootell, G., & Van Acker, A. (2004). *fib Bulletin 29. Precast concrete bridges*. fib. The International Federation for Structural Concrete. https://doi.org/10.35789/fib.BULL.0029

Camassa, D., Castellano, A., Fraddosio, A., & Piccioni, M. D. (2020). A New Ultrasonic Amplitude Tomography Approach, with Validation on Masonry Tuff Blocks. *Journal of Nondestructive Evaluation*, *39*(3). https://doi.org/10.1007/s10921-020-00693-2

Cheng, H., Weemstra, C., Hendriks, M. A. N., & Yang, Y. (2024). Comparing the stretching technique and the wavelet cross-spectrum technique for measuring stress-induced wave-velocity changes in concrete. *Automation in Construction*, *158*. https://doi.org/10.1016/j.autcon.2023.105221

Cheng, H., Zhang, F., Yang, Y., & Blom, C. B. M. (2023). Monitoring of repaired concrete floor in the Maastunnel using smart aggregates. *Bridge Safety, Maintenance, Management, Life-Cycle, Resilience and Sustainability - Proceedings of the 11th International Conference on Bridge Maintenance, Safety and Management, IABMAS 2022*, 493–500. https://doi.org/10.1201/9781003322641-57

Dall'Asta, A., & Zona, A. (2002). Non-linear analysis of composite beams by a displacement approach. *Computers & Structures*, *80*(27–30), 2217–2228. https://doi.org/10.1016/S0045-7949(02)00268-7

Davaadorj, O., Calvi, P. M., & Stanton, J. F. (2020). Shear Stress Transfer Across Concrete-to-Concrete Interfaces: Experimental Evidence and Available Strength Models. *PCI Journal*, *65*(4). https://doi.org/10.15554/pcij65.4-04

Debski, W., & Young, R. P. (1999). Enhanced velocity tomography: Practical method of combining velocity and attenuation parameters. *Geophysical Research Letters*, *26*(21), 3253–3256. https://doi.org/10.1029/1998gl010368

Farrar, C. R., & Worden, K. (2007). An introduction to structural health monitoring. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, *365*(1851), 303–315. https://doi.org/10.1098/rsta.2006.1928

Fröjd, P., & Ulriksen, P. (2017). Frequency selection for coda wave interferometry in concrete structures. *Ultrasonics*, *80*, 1–8. https://doi.org/10.1016/j.ultras.2017.04.012

Guan, S., Rice, J., Li, C., & Wang, G. (2014). Bridge Deflection Monitoring Using Small, Low-Cost Radar Sensors. *Structures Congress 2014*, 2853–2862. https://doi.org/10.1061/9780784413357.249

Haitsma Beton B.V. (n.d.). *Infra - Haitsma Beton*.

Haque, S., Eberhart, Z., Bansal, A., & McMillan, C. (2022). Semantic Similarity Metrics for Evaluating Source Code Summarization. *IEEE International Conference on Program Comprehension*, *2022-March*, 36–47. https://doi.org/10.1145/nnnnnnn.nnnnnnn

Hassani, F., Momayez, M., Tremblay, S., Momayez, M., Hassani, F. P., Guevremont, P., Saleh, K., & Tremblay, S. (2001). *A New Method for Testing Concrete in Dams An investigation of the MSR Impact-echo method for concrete dam inspections*. https://www.researchgate.net/publication/257129530

Hodson, T. O. (2022). Root-mean-square error (RMSE) or mean absolute error (MAE): when to use them or not. *Geoscientific Model Development*, *15*(14), 5481–5487. https://doi.org/10.5194/gmd-15-5481-2022

Hoult, N. A., Fidler, P. R. A., Hill, P. G., & Middleton, C. R. (2010). Long-Term Wireless Structural Health Monitoring of the Ferriby Road Bridge. *Journal of Bridge Engineering*, *15*(2), 153–159. https://doi.org/10.1061/(ASCE)BE.1943-5592.0000049

Ibrahim, M. S., Kostense, N. W., Poliotti, M., & Yang, Y. (2025). *Concrete strength prediction contest*. https://concrete-prediction-contest.tudelft.nl/index.php/home

Ibrahim, M. S., Yang, Y. ;, Roosen, M. A., & Hendriks, M. A. N. (2022). *Challenges on the shear behavior of existing continuous precast girder bridges*. https://www.openaccess.nl/en/you-share-we-take-care

Ibrahim, M., & Yang, Y. (2022). *Specimens preparation for the experiments on precast concrete bridge at intermediate support: Group 1*.

Jalinoos, F., Olson, L. D., Aouad, M. F., & Balch, A. H. (1995). *Acoustic tomography for Qualitative Non-destructive Evaluation (QNDE) of structural concrete using a new Ultrasonic scanner source*.

Kaar, P. H., Kriz, L. B., & Hognestad, E. (1960). PORTLAND CEMENT ASSOCIATION RESEARCH AND DEVELOPMENT LABORATORIES PRECAST-PRESTRESSED CONCRETE BRIDGES L PILOT TESTS OF CONTINUOUS GIRDERS. In *Journal of the PCA Research and Development Laboratories* (Vol. 2, Number 2).

Karaiskos, G., Deraemaeker, A., Aggelis, D. G., & Van Hemelrijck, D. (2015). Monitoring of concrete structures using the ultrasonic pulse velocity method. *Smart Materials and Structures*, *24*(11). https://doi.org/10.1088/0964-1726/24/11/113001

Leonhardt, F., & Walther, R. (1962). Shear Strength of Concrete Under Static and Repeated Loading. *ACI Journal Proceedings*, *59*(1), 1–30. https://www.scribd.com/document/395815347/Leonhardt-Walther-Contribution-to-the-Treatment-of-Shear-in-Reinforced-Concrete

Mardanshahi, A., Sreekumar, A., Yang, X., Barman, S. K., & Chronopoulos, D. (2025). Sensing Techniques for Structural Health Monitoring: A State-of-the-Art Review on Performance Criteria and New-Generation Technologies. In *Sensors* (Vol. 25, Number 5). Multidisciplinary Digital Publishing Institute (MDPI). https://doi.org/10.3390/s25051424

Mattock, A. H., & Hawkins, N. M. (1972). Shear transfer in reinforced concrete-RECENT RESEARCH. *Pci Journal*, *17(2)*, 55–75.

McHenry, D., & Mattock, A. H. (1960). Development of continuity in precast prestressed construction. *IABSE Congress Report*, (6), 399–412. https://doi.org/10.5169/seals-7067

Mirza, S. A., & Furlong, R. W. (1985). Design of Reinforced and Prestressed Concrete Inverted T Beams for Bridge Structures. *PCI Journal*, *30*(4), 112–136. https://doi.org/10.15554/pcij.07011985.112.136

Mohamad, M. E., & Ibrahim, I. S. (2015). Interface shear strength of concrete-to concrete bond with and without projecting steel reinforcement. *Jurnal Teknologi*, *75*(1), 169–172. https://doi.org/10.11113/jt.v75.3707

Niederleithinger, E., Wang, X., Herbrand, M., & Müller, M. (2018). Processing ultrasonic data by coda wave interferometry to monitor load tests of concrete beams. *Sensors (Switzerland)*, *18*(6). https://doi.org/10.3390/s18061971

Pan, B., Qian, K., Xie, H., & Asundi, A. (2009). Two-dimensional digital image correlation for in-plane displacement and strain measurement: A review. *Measurement Science and Technology*, *20*(6). https://doi.org/10.1088/0957-0233/20/6/062001

Pereira dos Santos, D., & Guilherme Haach, V. (2022). Generation of ultrasonic tomography from time-domain propagation spectrum. *Ultrasonics*, *120*. https://doi.org/10.1016/j.ultras.2021.106666

Perlin, L. P., & Pinto, R. C. de A. (2019). Use of network theory to improve the ultrasonic tomography in concrete. *Ultrasonics*, *96*, 185–195. https://doi.org/10.1016/j.ultras.2019.01.007

Planès, T., & Larose, E. (2013). A review of ultrasonic Coda Wave Interferometry in concrete. In *Cement and Concrete Research* (Vol. 53, pp. 248–255). Elsevier Ltd. https://doi.org/10.1016/j.cemconres.2013.07.009

Pruchnicki, P., & Opieliński, K. J. (2021). A Method of Precise Pulse Onset Determination Using the Akaike Information Criterion for Ultrasound Transmission Tomography. *Vibrations in Physical Systems*, *32*(1). https://doi.org/10.21008/j.0860-6897.2021.1.15

Randl, N. (2013). Design recommendations for interface shear transfer in fib Model Code 2010. *Structural Concrete*, *14*(3), 230–241. https://doi.org/10.1002/suco.201300003

Santos, P., & Júlio, E. (2010, June). *Assessment of the Shear Strength between Concrete Layers*.

Santos, P. M. D., Brito, E. N., & Júlio, S. (2014). Interface Shear Transfer on Composite Concrete Members. *Aci Structural Journal*, *111*, 113–122. https://www.researchgate.net/publication/267211456

Santos, P. M. D., & Júlio, E. N. B. S. (2012). A state-of-the-art review on shear-friction. In *Engineering Structures* (Vol. 45, pp. 435–448). https://doi.org/10.1016/j.engstruct.2012.06.036

Schreier, H., Orteu, J. J., & Sutton, M. A. (2009). Image correlation for shape, motion and deformation measurements: Basic concepts, theory and applications. In *Image Correlation for Shape, Motion and Deformation Measurements: Basic Concepts,Theory and Applications*. Springer US. https://doi.org/10.1007/978-0-387-78747-3

Singh, S., Sachan, A. K., & Shanker, R. (2022). A review on smart aggregate based structural health monitoring. *Materials Today: Proceedings*, *78*, 28–35. https://doi.org/10.1016/j.matpr.2022.10.246

Snieder, R. (2006). The theory of coda wave interferometry. *Pure and Applied Geophysics*, *163*(2–3), 455–473. https://doi.org/10.1007/s00024-005-0026-6

Song, G., Gu, H., & Mo, Y. L. (2008). Smart aggregates: Multi-functional sensors for concrete structures - A tutorial and a review. *Smart Materials and Structures*, *17*(3). https://doi.org/10.1088/0964-1726/17/3/033001

Spanbeton BV. (2014). Historical Development of Prestressed Bridge Elements in the Netherlands. *CPI Worldwide*. https://www.cpi-worldwide.com/journals/artikel/34569

Stähler, S. C., Sens-Schönfelder, C., & Niederleithinger, E. (2011). Monitoring stress changes in a concrete bridge with coda wave interferometry. *The Journal of the Acoustical Society of America*, *129*(4), 1945–1952. https://doi.org/10.1121/1.3553226

Taha, M. M. M., & Jia, Y. (2018). POST-TENSIONING THE CONNECTION REGION OF PRECAST POST-TENSIONED BRIDGE GIRDERS FOR CONTINUITY. *Stavební Obzor - Civil Engineering Journal*, *27*(1). https://doi.org/10.14311/CEJ.2018.01.0004

Takiguchi, T. (2019). Ultrasonic tomographic technique and its applications. *Applied Sciences (Switzerland)*, *9*(5). https://doi.org/10.3390/app9051005

Távora, C. G., Aguiar, P. R., Castro, B. A., Alexandre, F. A., Andreoli, A. L., & Bianchi, E. C. (2021). Hinkley criterion applied to detection and location of burn in grinding process. *The International Journal of Advanced Manufacturing Technology*, *113*(11–12), 3177–3188. https://doi.org/10.1007/s00170-021-06828-7

Vergoossen, R., van Eck, G.-J., & Jilissen, D. (2022). *Re-using existing prefabricated prestressed concrete girders in new bridges*. 554–561. https://doi.org/10.2749/prague.2022.0554

Walraven, J. C., & Reinhardt, H. W. (1981). Theory and Experiments on the Mechanical Behaviour of Cracks in Plain and Reinforced Concrete Subjected to Shear Loading. *HERON*, *26*, 26–68. https://resolver.tudelft.nl/uuid:3d68bd1a-465d-4590-b33c-7ede99bbc251

Yan, S., Ma, H., Li, P., Song, G., & Wu, J. (2017). Development and application of a structural health monitoring system based on wireless smart aggregates. *Sensors (Switzerland)*, *17*(7). https://doi.org/10.3390/s17071641

Zengqiu, Y., Wu, W., Xiao, L., Zhou, E., Cao, Z., Hua, J., & Wang, Y. (2024). Iterative Pulse–Echo Tomography for Ultrasonic Image Correction. *Sensors*, *24*(6). https://doi.org/10.3390/s24061895

Zhang, S., Cheng, H., & Yang, Y. (2022). *Monitoring of Prestressed Geopolymer Girders using Smart Aggregates and Fibre Optic Sensors*.

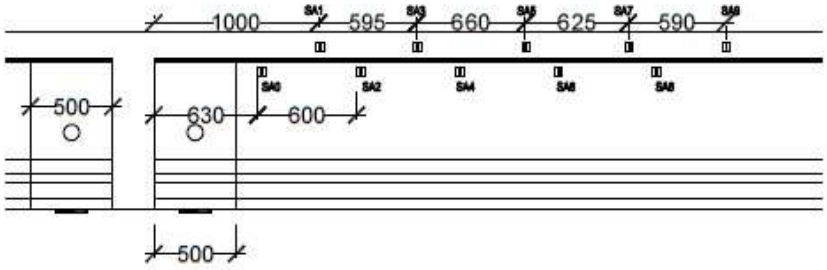
Zhang, Y., Abraham, O., Tournat, V., Le Duff, A., Lascoup, B., Loukili, A., Grondin, F., & Durand, O. (2013). Validation of a thermal bias control technique for Coda Wave Interferometry (CWI). *Ultrasonics*, *53*(3), 658–664. https://doi.org/10.1016/j.ultras.2012.08.003

Zhong, B., & Zhu, J. (2022). Applications of Stretching Technique and Time Window Effects on Ultrasonic Velocity Monitoring in Concrete. *Applied Sciences (Switzerland)*, *12*(14). https://doi.org/10.3390/app12147130

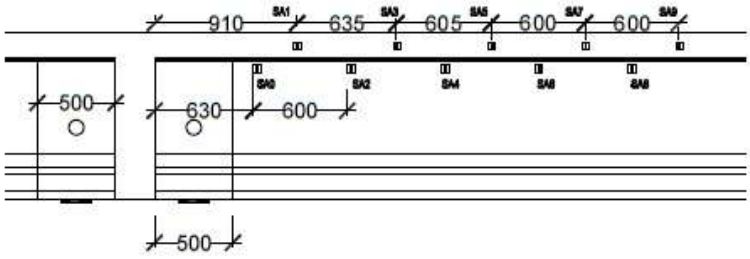
Appendix

Appendix I – Layout of the Smart Aggregates in the Beam

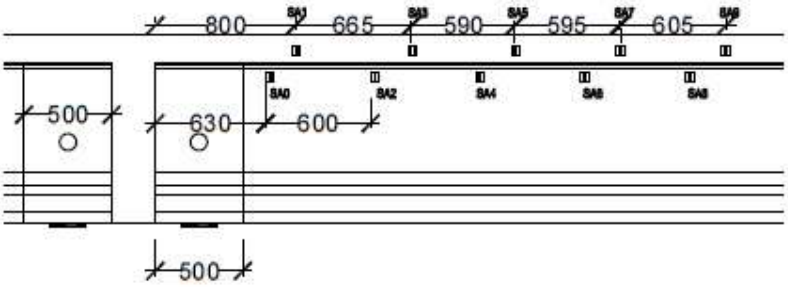
Smart Aggregate - S10H2D



Smart Aggregate - S10H2A



Smart Aggregate - S10H1B



Appendix II – Derivation and Restructuring of Key Formulas & Equations

Appendix II is organized as follows:

- **Appendix II.1** – overview of the system of equations
- **Appendix II.2** – restructuring of formula for ε_i
- **Appendix II.3** – algebraic process to determine the product functions
- **Appendix II.4** – sub-chapter on the unsuccessful tomographic translation of the CWI

Appendix II.1 – Overview of the System of Equations

All 25 equations used to express the system of equations for solving the tomography models are presented here, along with their respective Smart Aggregate (SA) pairings.

$$SA_0 \leftrightarrow SA_1, eq_1: t_1 = \frac{d_1}{v_2} + \frac{d_2}{v_1}$$
$$SA_0 \leftrightarrow SA_2, eq_2: t_2 = \frac{d_3}{v_2} + \frac{d_4}{v_4}$$
$$SA_0 \leftrightarrow SA_3, eq_3: t_3 = \frac{d_5}{v_2} + \frac{d_6}{v_4} + \frac{d_7}{v_3} + \frac{d_8}{v_5} + dt_1$$

$$SA_1 \leftrightarrow SA_2, eq_4: t_4 = \frac{d_9}{v_3} + \frac{d_{10}}{v_4} + dt_1$$
$$SA_1 \leftrightarrow SA_3, eq_5: t_5 = \frac{d_{11}}{v_3} + \frac{d_{12}}{v_5} +$$
$$SA_1 \leftrightarrow SA_4, eq_6: t_6 = \frac{d_{13}}{v_3} + \frac{d_{14}}{v_5} + \frac{d_{15}}{v_6} + \frac{d_{16}}{v_8} + dt_2$$

$$SA_2 \leftrightarrow SA_3, eq_7: t_7 = \frac{d_{17}}{v_6} + \frac{d_{18}}{v_5} + dt_2$$
$$SA_2 \leftrightarrow SA_4, eq_8: \ t_8 = \frac{d_{19}}{v_6} + \frac{d_{20}}{v_8}$$
$$SA_2 \leftrightarrow SA_5, eq_9: t_9 = \frac{d_{21}}{v_6} + \frac{d_{22}}{v_8} + \frac{d_{23}}{v_7} + \frac{d_{24}}{v_9} + dt_3$$

$$SA_3 \leftrightarrow SA_4, eq_{10}: t_{10} = \frac{d_{25}}{v_7} + \frac{d_{26}}{v_8} + dt_3$$
$$SA_3 \leftrightarrow SA_5, eq_{11}: t_{11} = \frac{d_{27}}{v_7} + \frac{d_{28}}{v_9}$$
$$SA_3 \leftrightarrow SA_6, eq_{12}: t_{12} = \frac{d_{29}}{v_7} + \frac{d_{30}}{v_9} + \frac{d_{31}}{v_{10}} + \frac{d_{32}}{v_{12}} + dt_4$$
$$SA_3 \leftrightarrow SA_8, eq_{13}: t_{13} = \frac{d_{33}}{v_7} + \frac{d_{34}}{v_9} + \frac{d_{35}}{v_{11}} + \frac{d_{36}}{v_{12}} + \frac{d_{37}}{v_{14}} + \frac{d_{38}}{v_{16}} + dt_5$$

$$SA_4 \leftrightarrow SA_5, eq_{13}: t_{14} = \frac{d_{39}}{v_{10}} + \frac{d_{40}}{v_9} dt_4$$
$$SA_4 \leftrightarrow SA_6, eq_{15}: t_{15} = \frac{d_{41}}{v_{10}} + \frac{d_{42}}{v_{12}}$$
$$SA_4 \leftrightarrow SA_7, eq_{16}: t_{16} = \frac{d_{43}}{v_{10}} + \frac{d_{44}}{v_{12}} + \frac{d_{45}}{v_{11}} + \frac{d_{46}}{v_{13}} + dt_5$$

$$SA_5 \leftrightarrow SA_6, eq_{17}: t_{17} = \frac{d_{47}}{v_{11}} + \frac{d_{48}}{v_{12}} + dt_5$$

$$SA_5 \leftrightarrow SA_7, eq_{18}: t_{18} = \frac{d_{49}}{v_{11}} + \frac{d_{50}}{v_{13}}$$

$$SA_5 \leftrightarrow SA_8, eq_{19}: t_{19} = \frac{d_{51}}{v_{11}} + \frac{d_{52}}{v_{13}} + \frac{d_{53}}{v_{14}} + \frac{d_{54}}{v_{16}} + dt_6$$

$$SA_6 \leftrightarrow SA_7, eq_{20}: t_{20} = \frac{d_{55}}{v_{14}} + \frac{d_{56}}{v_{13}} + dt_6$$

$$SA_6 \leftrightarrow SA_8, eq_{21}: t_{21} = \frac{d_{57}}{v_{14}} + \frac{d_{58}}{v_{16}}$$

$$SA_6 \leftrightarrow SA_9, eq_{22}: t_{22} = \frac{d_{59}}{v_{14}} + \frac{d_{60}}{v_{16}} + \frac{d_{61}}{v_{15}} + \frac{d_{62}}{v_{17}} + dt_7$$

$$SA_7 \leftrightarrow SA_8, eq_{23}: t_{23} = \frac{d_{63}}{v_{15}} + \frac{d_{64}}{v_{16}} + dt_7$$

$$SA_7 \leftrightarrow SA_9, eq_{24}: t_{24} = \frac{d_{65}}{v_{15}} + \frac{d_{66}}{v_{17}}$$

$$SA_8 \leftrightarrow SA_9, eq_{25}: t_{25} = \frac{d_{67}}{v_{18}} + \frac{d_{68}}{v_{17}}$$

Appendix II.2 – Tomographic reconstruction for CWI

UPV-based and CWI-based tomography calculations differ in how they derive the necessary input data –the arrival time ($t_{i,j}$)– to solve the system of equations. In UPV-based tomography, the arrival times are determined through the measured and the subsequent post-processed data and can then be directly used to solve for the wave velocities in each field.
However, CWI method does not provide arrival times; instead, it determines the change in relative velocity (ε). As the system of equations are based on the velocity method, it relies on arrival times to solve for the field velocities. Therefore, the gap between the CWI data must be bridged to obtain arrival times. This is done by first back calculating the arrival times using the ε factors obtained from CWI, using the formula here below:

$$t_{i,j} = t_{i,0} \Big/ \prod_{j=1}^{j \in loadsteps} \left(\varepsilon_{i,j} + 1\right) \tag{15}$$

where $t_{i,j}$is the arrival time of SA sensor pair i at load step j, and $t_{i,0}$ is the initial arrival time of the same sensor pair at load step 0. The factor $\prod(\varepsilon_{i,j} + 1)$ represents the cumulative influence of relative velocity changes and increases with each successive step, of the same SA sensor pair and at the current load step j.

This process begins with the initial arrival times at load step 0 ($t_{i,0}$), obtained from the UPV measurements. This arrival time is taken as a starting point, and as the arrival time for load step 0 for the CWI calculation. Then for each subsequent load step, the initial arrival time is adjusted by multiplying it with this cumulative product of relative velocity changes, represented as *prod_eps*, in the pseudo-code and $\prod(\varepsilon_{i,j} + 1)$ in the Formula (15). This reflects the compounding delay introduced by structural changes over time. Rather than updating from the previous step, each adjusted time value is consistently derived from the initial arrival time at load step 0 ($t_{i,0}$), modified by the full accumulation of velocity changes up to the current load step. This stepwise buildup is demonstrated in the pseudo-code provided here below.

Algorithm 1. Pseudo-code for the calculation of the arrival time through the relative velocity change (ε)

Pseudo-code arrival time

1. First loop, for each sensor pair index i:
    1.1 Initialize cumulative product of relative velocity change:
        prod_eps ← 1

2. Second loop within the first loop, for each load step j from 2 to the final load steps:
    2.1Retrieve ε_{ij} ← relative velocity change for pair i at load step j
    2.2 Update cumulative product:
        prod_eps ← *prod_eps* × (1 + ε_{ij})
    2.3 Compute adjusted arrival time:
        t_{ij} ← t_{i0} / *prod_eps*
    2.4 Store t_{ij} as the CWI-estimated arrival time at load step j, and *prod_eps* for the next load step

At the end of the calculation, after the *fmincon* algorithm has found the corresponding field velocities using the CWI input values, a final post-calculation is required to back-calculate the field velocities to the relative change in velocities ($\varepsilon_{k,j}$) for each field, using the following formula:

$$\varepsilon_{k,j} = \frac{v_{k,j}}{\prod_{j=1}^{j \in loadsteps} \left(\left(\varepsilon_{k,j-1} + 1 \right) v_{k,0} \right)} - 1 \qquad \text{(I)}$$

where $\varepsilon_{k,j}$ is the relative change in velocity of field k at load step j, $v_{k,j}$ is the field velocity at the same field and load step, $v_{k,0}$ is the initial velocity of that field, and the $\varepsilon_{k,j-1}$ is the cumulative change in the relative velocities factor of the same field k up to the previous load step ($j-1$).

Similar to the process describe here above to obtain the adjusted arrival times ($t_{i,j}$the procedure to back-calculate the relative change in velocity ($\varepsilon_{k,j}$) begins by retrieving the initial field velocity at load step 0 kN ($v_{k,0}$), and by initializing the cumulative product term—referred to as *prod_eps* in the pseudo-code and represented as $\prod(\varepsilon_{k,j-1} + 1)$ in the Formula (I) —to 1. Then, for each subsequent load step ($j \geq 2$), the field velocity ($v_{k,j}$) is divided by the product of the cumulative velocity change *prod_eps* and initial field velocity ($v_{k,0}$), to compute the new relative change ($\varepsilon_{k,j}$). After each step, the cumulative product is updated as carrying forward the accumulated changes. Each relative velocity change ($\varepsilon_{k,j}$) is therefore consistently derived from the initial velocity ($v_{k,0}$), adjusted by the cumulative effect of all preceding velocity changes up to the current load step. This iterative process is reflected in the following pseudo-code:

Algorithm 2. Pseudo-code for the calculation of the relative velocity change (ε) through the field velocity

Pseudo-code relative velocity change

1. First loop, for each velocity field index k:
    1.1 Retrieve v_{kj} ← velocity vector for field k
    1.2 Store initial velocity for field k:
        v_0 ← velocity [1]
    1.3 Initialize cumulative product:
        prod_eps ← 1

2. Second loop within the first loop, for each load step j from 2 to final load step:
    2.1 Compute relative velocity change:
        ε_{kj} ← (v_{kj} / (*prod_eps* × v_0)) – 1
    2.2 Update cumulative product:
        prod_eps ← *prod_eps* × (1 + ε_{kj})
    2.3 Store ε_{kj} as the CWI-based relative velocity change for field k at step j, and prod_eps for the next load step

An overview for all the formulas, and the algebraic steps taken to derive or rewriting the formulas seem in this chapter, can be found in Appendix II.3 – Formula for ε_i.

Appendix II.3 – Formula for ε_i

Formula for ε_i expressed in both the velocity and time domains

Algebraic process to rewrite the formula for relative change in velocity (ε_i) to be expressed in the time domain instead of velocity.

Step 1 – Stating the initial formula

Using the formula to determine the relative change in velocity (ε_i):

$$\varepsilon_i = \frac{dv}{v_{i-1}} = \frac{v_i - v_{i-1}}{v_{i-1}} \tag{I}$$

Step 2 – Introducing the relationship between velocity and time

Assuming a constant path length (l) across load steps, and using the relationship:

$$v = \frac{l}{t}$$

We substitute into (l) to yield the time-domain form:

$$\frac{dv}{v_{i-1}} = \frac{v_i - v_{i-1}}{v_{i-1}} = \frac{\frac{l}{t_i} - \frac{l}{t_{i-1}}}{\frac{l}{t_{i-1}}} = \frac{t_{i-1} - t_i}{t_i}$$

$$\rightarrow \frac{dt}{t_i} = \frac{t_{i-1} - t_i}{t_i} \tag{II}$$

Rewriting the Formulas (I) & (II)

The algebraic process to rewriting both Formulas (I) & (II) to be expressed as v_{i+1} & t_{i+1}, respectively:

$$\frac{dv}{v_{i-1}} = \frac{v_i - v_{i-1}}{v_{i-1}}$$

$$\rightarrow \varepsilon_i = \frac{v_i - v_{i-1}}{v_{i-1}}$$

$$\rightarrow \varepsilon_i = \frac{v_i}{v_{i-1}} - \frac{v_{i-1}}{v_{i-1}}$$

$$\rightarrow \varepsilon_i = \frac{v_i}{v_{i-1}} - 1$$

$$\rightarrow \varepsilon_i + 1 = \frac{v_i}{v_{i-1}}$$

$$\rightarrow v_i = (\varepsilon_i + 1) v_{i-1} \tag{III}$$

$$\rightarrow (\varepsilon_i + 1) = \frac{v_i}{v_{i-1}} \tag{IV}$$

And for the time domain:

$$\frac{dt}{t} = \frac{t_{i-1} - t_i}{t_i}$$

$$\rightarrow \varepsilon_i = \frac{t_{i-1} - t_i}{t_i}$$

$$\rightarrow \varepsilon_i = \frac{t_{i-1}}{t_i} - \frac{t_i}{t_i}$$

$$\rightarrow \varepsilon_i = \frac{t_{i-1}}{t_i} - 1$$

$$\rightarrow \varepsilon_i + 1 = \frac{t_{i-1}}{t_i}$$

$$\rightarrow t_i = \frac{t_{i-1}}{(\varepsilon_i + 1)} \tag{V}$$

$$\rightarrow (\varepsilon_i + 1) = \frac{t_{i-1}}{t_i} \tag{VI}$$

Appendix II.4 – The algebraic Process to determine the Product Functions

The main goal is to link the base Formula (IV), which determines the relative velocity between two subsequent load steps, and alter it so that all the subsequent load steps and their relative change in velocity (ε_i) are linked to the initial velocity (v_0), at load step 0 kN instead of to the load step before it. A step-by-step example is given below:

Velocity Domain Version

Step 1 – Determine the initial formula, at load step 50 kN

From the Formulas (IV):

$$(\varepsilon_1 + 1) = \frac{v_1}{v_0} \tag{VII}$$

Step 2 - Determine the formula, at load step 100 kN

From the Formulas (IV):

$$(\varepsilon_2 + 1) = \frac{v_2}{v_1} \tag{VIII}$$

Multiply Formula (VII) with (VIII):

$$(\varepsilon_1 + 1) = \frac{v_1}{v_0} \;\&\; (\varepsilon_2 + 1) = \frac{v_2}{v_1}$$

$$\rightarrow (\varepsilon_1 + 1) \cdot (\varepsilon_2 + 1) = \frac{v_1}{v_0} \cdot \frac{v_2}{v_1}$$

$$\rightarrow (\varepsilon_1 + 1) \cdot (\varepsilon_2 + 1) = \frac{v_2}{v_0} \tag{IX}$$

$$\rightarrow v_2 = (\varepsilon_1 + 1) \cdot (\varepsilon_2 + 1) v_0 \tag{X}$$

Step 3 – Determine the formula, at load step 150 kN

Repeat the process using Formula (IX):

$$(\varepsilon_1 + 1) \cdot (\varepsilon_2 + 1) = \frac{v_2}{v_0} \;\&\; (\varepsilon_3 + 1) = \frac{v_3}{v_2}$$

$$\rightarrow (\varepsilon_1 + 1) \cdot (\varepsilon_2 + 1) \cdot (\varepsilon_3 + 1) = \frac{v_2}{v_0} \cdot \frac{v_3}{v_2}$$

$$\rightarrow (\varepsilon_1 + 1) \cdot (\varepsilon_2 + 1) \cdot (\varepsilon_3 + 1) = \frac{v_3}{v_0} \tag{XI}$$

$$\rightarrow v_3 = (\varepsilon_1 + 1) \cdot (\varepsilon_2 + 1) \cdot (\varepsilon_3 + 1) v_0 \tag{XII}$$

Step 4 – Determining the product function

A sequence can be seen in how the formula evolves at each load step. A general product function for v_i can be derived:

$$v_i = \prod_{i=1}^{i \in load\ steps} (\varepsilon_i + 1)\, v_0 \tag{XIII}$$

Step 5 – Rewrite to express ε_i:

$$v_i = \prod_{i=1}^{i \in loadsteps} (\varepsilon_i + 1)\, v_0$$

$$\rightarrow \prod_{i=1}^{i \in loadsteps} (\varepsilon_i + 1) = \frac{v_i}{v_0}$$

$$\rightarrow (\varepsilon_i + 1) \cdot \prod_{i=1}^{i \in loadsteps} (\varepsilon_i + 1) = \frac{v_i}{v_0}$$

$$\rightarrow (\varepsilon_i + 1) = \frac{v_i}{\prod_{i=1}^{i \in loadsteps} (\varepsilon_i + 1) \cdot v_0}$$

$$\rightarrow \varepsilon_i = \frac{v_i}{\prod_{i=1}^{i \in loadsteps} (\varepsilon_i + 1) \cdot v_0} - 1 \qquad \text{(XIV)}$$

Formula (XIV)is the equivalent to Formula (**Error! Reference source not found.**).

Time Domain Version

Step 1 – Determine the initial formula at load step 50 kN

From Formulas (V):

$$(\varepsilon_1 + 1) = \frac{t_0}{t_1} \qquad \text{(XV)}$$

Step 2 - Determine the formula at load step 100 kN

From Formulas (V):

$$(\varepsilon_2 + 1) = \frac{t_1}{t_2} \qquad \text{(XVI)}$$

Multiply Formula (XV) with (XVI):

$$(\varepsilon_1 + 1) = \frac{t_0}{t_1} \;\&\; (\varepsilon_2 + 1) = \frac{t_1}{t_2}$$

$$\rightarrow\ (\varepsilon_1 + 1) \cdot (\varepsilon_2 + 1) = \frac{t_0}{t_1} \cdot \frac{t_1}{t_2}$$

$$\rightarrow\ (\varepsilon_1 + 1) \cdot (\varepsilon_2 + 1) = \frac{t_0}{t_2} \qquad \text{(XVII)}$$

$$\rightarrow t_2 = \frac{t_0}{(\varepsilon_1 + 1) \cdot (\varepsilon_2 + 1)} \qquad \text{(XVIII)}$$

Step 3 – Determine the formula at load step 150 kN

Repeat the process using Formula (XVII):

$$(\varepsilon_1 + 1) \cdot (\varepsilon_2 + 1) = \frac{t_0}{t_2} \;\&\; (\varepsilon_3 + 1) = \frac{t_2}{t_3}$$

$$\rightarrow\ (\varepsilon_1 + 1) \cdot (\varepsilon_2 + 1) \cdot (\varepsilon_3 + 1) = \frac{t_0}{t_2} \cdot \frac{t_2}{t_3}$$

$$\rightarrow\ (\varepsilon_1 + 1) \cdot (\varepsilon_2 + 1) \cdot (\varepsilon_3 + 1) = \frac{t_0}{t_3} \qquad \text{(XIX)}$$

$$\rightarrow t_3 = \frac{t_0}{(\varepsilon_1 + 1) \cdot (\varepsilon_2 + 1) \cdot (\varepsilon_3 + 1)} \qquad \text{(XX)}$$

Step 4 – Determining the product function

General expression for t_i:

$$t_i = t_0 / \prod_{i=1}^{i \in loadsteps} (\varepsilon_i + 1) \tag{XXI}$$

Step 5 – Rewrite to express ε_i:

$$t_i = t_0 / \prod_{i=1}^{i \in loadsteps} (\varepsilon_i + 1)$$

$$\rightarrow \prod_{i=1}^{i \in loadsteps} (\varepsilon_i + 1) = \frac{t_0}{t_i}$$

$$\rightarrow (\varepsilon_i + 1) \cdot \prod_{i=1}^{i \in loadsteps} (\varepsilon_{i-1} + 1) = \frac{t_0}{t_i}$$

$$\rightarrow (\varepsilon_i + 1) = \frac{t_0}{t_i \prod_{i=1}^{i \in loadsteps} (\varepsilon_{i-1} + 1)}$$

$$\rightarrow \varepsilon_i = \frac{t_0}{\prod_{i=1}^{i \in loadsteps} (\varepsilon_{i-1} + 1) \cdot t_i} - 1 \tag{XXII}$$

Appendix III – MATLAB code

Note: All MATLAB code examples provided in this appendix are based on the test beam **S10H2D**. Input variable names and the number of variables may differ slightly for other beams; however, the overall structure and logic of the code remain consistent.
Appendix III is organized as follows:

- **Appendix III.1** – Code for initial raw data sorting
- **Appendix III.2** – Code used for the UPV (Ultrasonic Pulse Velocity) method
- **Appendix III.3** – Code used for the CWI (Coda Wave Interferometry) method
- **Appendix III.4** – Algorithms and processing steps for the Tomography method
- **Appendix III.5** – Code used to generate Ray Path plots
- **Appendix III.6** – Code used to generate Tomographic plots
- **Appendix III.7** – Code used to adjust and modify the DIC observations

Appendix III.1 – Data Sorting Code

Appendix III.1.1 – raw_data_sort.m

v1.1 for Mohammed girder test on 2022-11-21

written by Hao Cheng extract the data from tests into a MATLAB structure please run the code in the RAW DATA folder

input

```
tic;

all_source=[0;1;2;3;4;5;6;7;8;9]; % all sources dont need change
sensor=[0;8;1;9;2;3;4;5;6;7]; % all sensors dont need change
% channel=[3;4;8;9;13;18;23;28;33;38]; % corresponding channels for sensors, Long version
RawData
channel=[3;4;5;6;7;9;11;13;15;17]; % corresponding channels for sensors, Short version
RawData

num_of_load_step=19; % CHANGE IT BEFORE RUNNING

folder_path='C:\Users\Vincent Ponson\Documents\01- TUDelft\02 - MSc\Courses\02 - Masters
Thesis\07 - Data Test\061 - 2023-03-21 Test\'; % change this to your folder path
file_name=dir([folder_path '*.csv']); % dont need change
file=[1;num_of_load_step+1;2*num_of_load_step+1;3*num_of_load_step+1;4*num_of_load_step+1;5*n
um_of_load_step+1;...
    6*num_of_load_step+1;7*num_of_load_step+1;8*num_of_load_step+1;9*num_of_load_step+1]; %
file number: depends on the load steps
k=1; % row in Matlab structure file dont need change
```

data processing

```
for j=1:length(all_source)

    source_sensor=all_source(j);
    file_to_read=(num_of_load_step*(j-1)+1):(num_of_load_step*j); % number of files\number of
load steps

    pos=find(sensor==source_sensor);
    source_channel=channel(pos);

    sensor_use=setxor(source_sensor,sensor);
```

```
    channel_use=setxor(source_channel,channel);

    for i=1:length(file_to_read)
        rawdata_tem=readmatrix([file_name(file_to_read(i)).name]);
        rawdata{i,1}=rawdata_tem;
    end

    for i=1:length(sensor_use)
        pos_tem=channel(find(sensor==sensor_use(i)));
        data(k).source=source_sensor;
        data(k).receiver=sensor_use(i);
        data(k).load_0kN_before_test=rawdata{1}(2:end,pos_tem);
        data(k).load_50kN=rawdata{2}(2:end,pos_tem);
        data(k).load_100kN=rawdata{3}(2:end,pos_tem);
        data(k).load_150kN=rawdata{4}(2:end,pos_tem);
        data(k).load_200kN=rawdata{5}(2:end,pos_tem);
        data(k).load_250kN=rawdata{6}(2:end,pos_tem);
        data(k).load_300kN=rawdata{7}(2:end,pos_tem);
        data(k).load_350kN=rawdata{8}(2:end,pos_tem);
        data(k).load_400kN=rawdata{9}(2:end,pos_tem);
        data(k).load_450kN=rawdata{10}(2:end,pos_tem);
        data(k).load_500kN=rawdata{11}(2:end,pos_tem);
        data(k).load_550kN=rawdata{12}(2:end,pos_tem);
        data(k).load_600kN=rawdata{13}(2:end,pos_tem);
        data(k).load_650kN=rawdata{14}(2:end,pos_tem);
        data(k).load_700kN=rawdata{15}(2:end,pos_tem);
        data(k).load_750kN=rawdata{16}(2:end,pos_tem);
        data(k).load_800kN=rawdata{17}(2:end,pos_tem);
        data(k).load_850kN=rawdata{18}(2:end,pos_tem);
        data(k).load_900kN=rawdata{19}(2:end,pos_tem);
        data(k).load_0kN_after_test=rawdata{1}(2:end,pos_tem);
        k=k+1;

        % Change list of data to correspond to the total amount of loadstpes
    end

end

toc;
```

Published with MATLAB® R2022b

Appendix III.2 – UPV Code

Appendix III.2.1 – using_obtain_velocity.m

V1.2 for Mohammed girder test on 2022-11-21 written by Hao Cheng

code

```
tic;
file_num=1;
sensor=[0;1;2;3;4;5;6;7;8;9];

for i=1:length(sensor)
    source = sensor(i);
    receiver = setxor(source,sensor);
```

```
    for j=1:length(receiver)
        velocity_tem = obtain_velocity(source,receiver(j),data,geometry);

        velocity(file_num).source = source;
        velocity(file_num).receiver = receiver(j);
        velocity(file_num).velocity = velocity_tem;

        file_num = file_num + 1;
    end
end
toc;

i=1;
j=2;
receiver = receiver(j);
```

END

Published with MATLAB® R2022b

Appendix III.2.2 – obtain_velocity.m

```
function [velocity]=obtain_velocity(source,receiver,data,geometry)
```

v1.2 for Mohammed girder test on 2022-11-21

written by Hao Cheng (h.cheng-2@tudelft.nl) for obtaining UPV using AIC and Hinkley

preparation

```
pos=find([data.source]==source&[data.receiver]==receiver);
sig=[data(pos).load_0kN_before_test data(pos).load_50kN data(pos).load_100kN ...
    data(pos).load_150kN data(pos).load_200kN data(pos).load_250kN ...
    data(pos).load_300kN data(pos).load_350kN data(pos).load_400kN ...
    data(pos).load_450kN data(pos).load_500kN data(pos).load_550kN ...
    data(pos).load_600kN data(pos).load_650kN data(pos).load_700kN ...
    data(pos).load_750kN data(pos).load_800kN data(pos).load_850kN ...
    data(pos).load_900kN data(pos).load_950kN data(pos).load_1000kN ...
    data(pos).load_1050kN data(pos).load_1100kN data(pos).load_1150kN ...
    data(pos).load_1200kN data(pos).load_1250kN data(pos).load_1300kN ...
    data(pos).load_1350kN data(pos).load_1400kN data(pos).load_1450kN ...
    data(pos).load_1500kN]; % change
```

offset compensation and interpolation

```
for j=1:size(sig,2)
%     sig(:,j)=interpolation(sig(:,j));
    sig(:,j) = sig(:,j)-mean(sig(180:end,j));
end
```

time and frequency

```
fs  = 3e6;
t   = 1/fs:1/fs:size(sig,1)/fs;
```

```
tu  = t';
dtu = tu(2)-tu(1);
```

direct wave velocity check

```
pos_source   = find([geometry.sensor]==source);
pos_receiver = find([geometry.sensor]==receiver);
distance     = sqrt((geometry(pos_source).height - geometry(pos_receiver).height)^2+...
    (geometry(pos_source).distance_from_edge-geometry(pos_receiver).distance_from_edge)^2);

for i=1:size(sig,2)
    w1       = find(tu>30e-6); %initial time where we look
    [~,t0,~] = Hinkley_pick(sig(w1,i),tu(w1));

    w2               = find(tu>t0-30e-6 & tu<t0+30e-6);
    [arrival_time,~] = AIC_pick(sig(w2,i),fs);

    velocity(i) = distance./(arrival_time + t0 - 30e-6);
end
```

Published with MATLAB® R2022b

Appendix III.2.3 – Hinkley_pick.m

```
function [Hin,arvt,pos] = Hinkley_pick(sig,t)
```

Hinkley criterion

Writen by Fengqiao Zhang 2018 Adjusted by Vincent Ponson (v.n.ponson@student.tudelft.nl)

code

```
sig = sig/max(abs(sig));% normalized
N   = length(sig);

for ii     = 1:N
    Si(ii) = sum(sig(1:ii).^2);
end

alpha = 50;
Sn    = Si(end);
delta = Sn/(alpha*N);
Hin   = Si-(1:N)*delta;
pos   = find(Hin==min(Hin));
arvt  = t(pos);

end
```

END

Published with MATLAB® R2022b

Appendix III.2.4 – AIC_pick.m

```
function [arrival_time,aic]=AIC_pick(x,fs)
```

v1.2 for Mohammed girder test on 2022-11-21
written by Hao Cheng (h.cheng-2@tudelft.nl) for interpolating clipping signal

code

```
N=length(x);
for i=1:N
    aic(i) = i*log10(var(x(1:i))) + (N-i-1)*log10(var(x(i+1:end)));
end
arrival_time = find(aic==min(aic(21:length(x)-10)))/fs;

end
```

Published with MATLAB® R2022b

Appendix III.2.5 – Interpolation.m

```
function [sig_final]=interpolation(x)
```

v1.2 for Mohammed girder test on 2022-11-21
written by Hao Cheng (h.cheng-2@tudelft.nl) for interpolating clipping signal

code

```
pos_ori=1:length(x);
pos_ori=pos_ori';

pos_wrong1=find(x>3.4);
pos_wrong2=find(x<-3.4);
pos_wrong3=[pos_wrong1;pos_wrong2];
pos_wrong=sort(pos_wrong3);

pos_right=setxor(pos_ori,pos_wrong);
sig_right=x(pos_right,1);

synt=interp1(pos_right,sig_right,pos_wrong,'spline');

sig_final=zeros(length(x),1);
sig_final(pos_right)=sig_right;
sig_final(pos_wrong)=synt;

end
```

Published with MATLAB® R2022b

Appendix III.2.6 – Velocity_correction_V2.m & input_corr.m

Appendix III.2.6.1 – Velocity_correction_V2.m

v2.0 for Mohammed girder test on 2022-11-21
UPV correction code p1v2 written by Vincent Ponson (v.n.ponson@student.tudelft.nl) Correct errors in the picked point using UPV method

Manual Input

```
source   = [0,1,0,2,0, 1,2,1,3,1,4, 2,3,2,4,2,5, 3,4,3,5,3,6, 4,5,4,6,4,7, 5,6,5,7,5,8,
7,6,9];
```

```
receiver = [1,0,2,0,3, 2,1,3,1,4,1, 0,2,1,2,3,2, 4,3,5,3,6,3, 5,4,6,4,7,4, 6,5,7,5,8,5,
6,9,6];
```

Code

```
fs = 3e6;

for j = 1:length(source)
    loadstep = corr(j).loadstep;
    pos = find([velocity.source] == source(j) & [velocity.receiver] == receiver(j));

    pos_location_source = find([geometry.sensor] == source(j));
    pos_location_receiver = find([geometry.sensor] == receiver(j));

    for i = 1:length(loadstep)
        distance = sqrt((geometry(pos_location_source).distance_from_edge -
geometry(pos_location_receiver).distance_from_edge)^2 + ...
            (geometry(pos_location_source).height -
geometry(pos_location_receiver).height)^2);
        picked_time = corr(j).picked_point(i) / fs;
        real_velocity = distance / picked_time;

        % Overwrite the velocity at the i-th Loadstep for the given position
        velocity(pos).velocity(loadstep(i)) = real_velocity;
    end
end
```

Published with MATLAB® R2022b

Appendix III.2.6.2 – input_corr.m

v1.0 for Mohammed girder test on 2022-11-21

UPV correction code p2v2 written by Vincent Ponson (v.n.ponson@student.tudelft.nl) Creat struct for correcting errors in the velocity calculations

Input

Define the Loadstep & picked_points list

```
loadstep = {
[12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27];
[17,18,19,20,21,22,23,24,25,26,27];
[1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28];
[1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28];
[8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27];
};

picked_point = {
[368,368,368,368,368,368,368,368,368,368,368,368,368,368,368,368];
[270,270,270,270,270,270,270,270,270,270,270];
[406,407,409,411,413,416,420,421,425,426,550,600,625,625,625,625,625,625,625,625,625,625,
625,625,625,625,406];
[405,409,415,420,427,438,440,475,480,618,625,625,625,625,625,625,625,625,625,625,625,625,
625,625,625,625,405];
[583,594,620,806,825,825,825,825,825,825,825,825,825,825,825,825,825,825,825,825,825];
};
```

```
for i=1:length(loadstep)
    a = loadstep{i};
    b = picked_point{i};
    corr(i).loadstep = a;
    corr(i).picked_point = b;
    corr(i).check = length(a) - length(b);
end
```

Published with MATLAB® R2022b

Appendix III.3 – CWI Code

Appendix III.3.1 – using_ stretching.m

v1.2 for Mohammed girder test on 2022-11-21
written by Hao Cheng (h.cheng-2@tudelft.nl)

```
tic;
f=waitbar(0,'Please wait...');
load Data_30-03 %change name to RawData filesname
load girder_bending_S10H2D.mat %change name to girder name S10Hxx
sensor=[0;1;2;3;4;5;6;7;8;9];
file_num=1;
waitbar(0,f,'Data processing...');
for i=1:length(sensor)
    source=sensor(i);
    receiver=setxor(source,sensor);
    for j=1:length(receiver)
        [EPS,CC,corr_coef]=stretching(source,receiver(j),data,geometry);
        DWI(file_num).source=source;
        DWI(file_num).receiver=receiver(j);
        DWI(file_num).EPS=EPS;
        DWI(file_num).CC=CC;
        DWI(file_num).corr_coef=corr_coef;
        file_num=file_num+1;
    end
    waitbar(i/length(sensor),f,'Data processing...');
end
toc;
```

Published with MATLAB® R2022b

Appendix III.3.2 – stretching.m

```
function [EPS,CC,corr_coef]=stretching(source,receiver,data,geometry)
```

v1.2 for Mohammed girder test on 2022-11-21
written by Hao Cheng (h.cheng-2@tudelft.nl) for selecting source and receiver

code

```
pos=find([data.source]==source&[data.receiver]==receiver);
sig=[data(pos).load_0kN_before_test data(pos).load_50kN data(pos).load_100kN ...
    data(pos).load_150kN data(pos).load_200kN data(pos).load_250kN ...
    data(pos).load_300kN data(pos).load_350kN data(pos).load_400kN ...
```

```
    data(pos).load_450kN data(pos).load_500kN data(pos).load_550kN ...
    data(pos).load_600kN data(pos).load_650kN data(pos).load_700kN ...
    data(pos).load_750kN]; %change to actual number of loadsteps

% for j=1:size(sig,2)
%     sig(:,j)=interpolation(sig(:,j));
%     sig(:,j)=sig(:,j)-mean(sig(70:80,j));
% end

fs=3e6; %fs= sampleing rate, Rawsignal 3e-6
t=1/fs:1/fs:size(sig,1)/fs;
tu=t';
dtu=tu(2)-tu(1);

for i=1:size(sig,2)-1
    ref=sig(:,i);
    sign=sig(:,i+1);
    [coef_corr,cc,eps]=stretching_calculation(ref,sign,tu,dtu);
    EPS(:,i)=eps;
    CC(:,i)=cc;
    corr_coef(:,i)=coef_corr;
end

end
```

Published with MATLAB® R2022b

Appendix III.3.3 – stretching_calcualtion.m

function [coef_corr,cc,eps]=stretching_calculation(ref,sign,tu,dtu)

v1.2 for Mohammed girder test on 2022-11-21

written by Hao Cheng (h.cheng-2@tudelft.nl) for performing signal stretching

parameters

```
id=find(tu>60e-6);
[~,t0,~] = Hinkley_pick(sign(id),tu(id),0);
epsilon=-2e-2:1e-5:5e-3;
```

stretching

```
time_window=round(t0/dtu):round((t0+200e-6)/dtu);

for EPS_index=1:length(epsilon)
    time2=(tu)*(1+epsilon(EPS_index));
    synt=interp1(tu,ref,time2,'spline');
    for DATE_index=1:size(sign,2)
        temp=corrcoef(synt(time_window),sign(time_window,DATE_index));
        COEF_CORR(EPS_index,DATE_index)=temp(2);
    end
end

coef_corr(:,1)=COEF_CORR;
[cc b]=max(COEF_CORR);
eps=epsilon(b)';
```

```
end
```

END

Published with MATLAB® R2022b

Appendix III.3.4 – Hinkley_pick.m

```
function [Hin,arvt,pos] = Hinkley_pick(sig,t,fign)
% Hinkley criterion
% Fengqiao Zhang 2018
sig        = sig/max(abs(sig));% normalized

if fign
    figure(fign);clf;hold on;
    plot(t*1e6,sig,'r-');
    xlabel('time [\mus]');
    ylabel('amplitude [V]');
    xlim([0 t(max(find(sig>0)))*1e6+50]);
    ylim([-1 1]);
    grid minor
    set(gca,'box','on');
end

N          = length(sig);
for ii     = 1:N
    Si(ii) = sum(sig(1:ii).^2);
end
alpha      = 50;
Sn           = Si(end);
delta       = Sn/(alpha*N);
Hin        = Si-(1:N)*delta;
pos       = find(Hin==min(Hin));
arvt       = t(pos);

if fign
    plot([arvt arvt]*1e6,[-1 1],'k-x');
end
end
```

Published with MATLAB® R2022b

Appendix III.4 – Tomography Code

Appendix III.4.1 – Data Sorting

Appendix III.4.1.1 – Tomo_Data_Sorter.m

VNPonson Tomography Graphs - V2.0

initial Tomography Code p1v5 - Tomography Data Sorter p1v3 written by Vincent Ponson (v.n.ponson@student.tudelft.nl) Setting up the Struct file - needed inputs for later Calculations (CWI)

```
%  Files needed:  CWI-data, UPV-data, Geometry
%  File createrd: TMD-CWI
```

Setting up Indexes and Preallocate memory

```
%Source - Receiver Time Pairing
source   = [0,0,0, 1,1,1,1, 2,2,2,2,2, 3,3,3,3,3,3,3, 4,4,4,4,4,4, 5,5,5,5,5,5, 6,6,6,6,6,6,
7,7,7,7,7, 8,8,8,8,8, 9,9,9];
receiver = [1,2,3, 0,2,3,4, 0,1,3,4,5, 0,1,2,4,5,6,8, 1,2,3,5,6,7, 2,3,4,6,7,8, 3,4,5,7,8,9,
4,5,6,8,9, 3,5,6,7,9, 6,7,8];
```

Main sorting

```
for i=1:length(source)
    % Find the positions of the source and receiver in the geometry array
    pos_g1 = find([geometry.sensor] == source(i));
    pos_g2 = find([geometry.sensor] == receiver(i));

    % Calculate the Height and Distance between the source and receiver
    hor_l   = abs(geometry(pos_g2).distance_from_edge - geometry(pos_g1).distance_from_edge);
    ver_l   = abs(geometry(pos_g2).height - geometry(pos_g1).height);
    llength = sqrt((geometry(pos_g1).distance_from_edge -
geometry(pos_g2).distance_from_edge)^2 + ...
    (geometry(pos_g1).height - geometry(pos_g2).height)^2);

    % Calculate the angle between the source and receiver
    angle = abs(atan((geometry(pos_g2).height - geometry(pos_g1).height)/ ...
            (geometry(pos_g2).distance_from_edge - geometry(pos_g1).distance_from_edge)));

    % Find the position in the velocity array that matches the source and receiver
    pos_v1 = find([velocity.source] == source(i) & [velocity.receiver] == receiver(i));
    vel = velocity(pos_v1).velocity;

    % Calculate (back) the Arrival time
    for j=1:length(vel)
        A_t(j) = llength / vel(j);
    end

     % Find the position in the velocity array that matches the source and receiver
    pos_v1 = find([DWI.source] == source(i) & [DWI.receiver] == receiver(i));
    eps = DWI(pos_v1).EPS;

    Tomo(i).source     = source(i);
    Tomo(i).receiver   = receiver(i);
    Tomo(i).hor_length = hor_l;
    Tomo(i).ver_length = ver_l;
    Tomo(i).SR_length  = llength;
    Tomo(i).angle      = angle;
    Tomo(i).velocity   = vel;
    Tomo(i).At         = A_t;
    Tomo(i).EPS        = eps;
end
```

Published with MATLAB® R2022b

Appendix III.4.1.2 – Time_calc.m

VNPonson Tomography Graphs - V2.0

initial Tomography Code p2v5 - Tomography Data Sorter p2v3 written by Vincent Ponson (v.n.ponson@student.tudelft.nl) Setting up the Struct file - average Travel Time(UPV&CWI) & EPS(CWI)

```
% Files needed:  TMD_(CWI/UPV)
% Files created: Time(t), EPS, Time_eps(te)
```

Setting up Indexes and Preallocate memory

Source - Receiver Time Pairing

```
source   =
[0,0,0,1,1,1,1,2,2,2,2,2,3,3,3,3,3,3,3,4,4,4,4,4,4,5,5,5,5,5,5,6,6,6,6,6,6,7,7,7,7,7,8,8,8,8,
8,9,9,9];
receiver =
[1,2,3,0,2,3,4,0,1,3,4,5,0,1,2,4,5,6,8,1,2,3,5,6,7,2,3,4,6,7,8,3,4,5,7,8,9,4,5,6,8,9,3,5,6,7,
9,6,7,8];

% Indexes to extract specific time steps
time_indices = [1,2,3,5,6,7,10,11,12,16,17,18,19,23,24,25,29,30,31,35,36,37,41,42,47];

% Preset loadstep count, CHANGE
loadsteps_UPV = 28; %UPV calc
loadsteps_CWI = 13; %CWI calc

% Preallocate memory for 't' and 'te' structures
t    = struct();
te   = struct();
t_av = zeros(length(source), loadsteps_UPV);

% Preallocate memory for 'eps' and 'EPS' structures
eps = zeros(length(source), loadsteps_CWI);
EPS = struct();
```

UPV Time calculations

```
for i = 1:loadsteps_UPV
    for j = 1:length(source)
        pos1 = find([Tomo.source] == source(j) & [Tomo.receiver] == receiver(j));
        pos2 = find([Tomo.source] == receiver(j) & [Tomo.receiver] == source(j));
        At1 = Tomo(pos1).At;
        At2 = Tomo(pos2).At;

        t_av(j, i) = (At1(i) + At2(i)) / 2;
    end
end

% Assign values from selected time indices to Time structure
for k = 1:length(time_indices)
    t.(['t', num2str(k)])  = t_av(time_indices(k), :); %UPV, initial time
    te.(['t', num2str(k)]) = [t_av(time_indices(k), 1), zeros(1,(loadsteps_CWI-1))];
end
```

Average EPS calc

```
for i = 1:loadsteps_CWI
    for j = 1:length(source)
        pos1 = find([Tomo.source] == source(j) & [Tomo.receiver] == receiver(j));
        pos2 = find([Tomo.source] == receiver(j) & [Tomo.receiver] == source(j));
        eps1 = Tomo(pos1).EPS;
        eps2 = Tomo(pos2).EPS;

        eps(j, i) = (eps1(i) + eps2(i)) / 2;
    end
end

% Assign values from selected EPS indices to EPS structure
for k = 1:length(time_indices)
    EPS.(['eps', num2str(k)]) = eps(time_indices(k), :);
end
```

CWI Time calculation

```
for i=1:length(time_indices)
    prod_eps = 1; % Reset cumulative product for each 't'
    for j=2:loadsteps_CWI
            current_eps = EPS.(sprintf('eps%d', i))(j);
            prod_eps = prod_eps * (current_eps + 1); % Cumulative product

            te.(sprintf('t%d', i))(j) = te.(sprintf('t%d', i))(1) / prod_eps; %CWI initial
time
    end
end
```

Published with MATLAB® R2022b

Appendix III.4.1.3 – Distance_calc.m

VNPonson Tomography Graphs - V2.0

initial Tomography Code p3v5 - Tomography Data Sorter p3v3 written by Vincent Ponson (v.n.ponson@student.tudelft.nl) Setting up the Struct file - the distance of each cell (UPV&CWI)

```
% Files needed:  TMD_(CWI/UPV), Geometry
% Files created: Distance
```

Setting up Indexes and Preallocate memory

Source - Receiver Distance Pairing

```
source   = [0,0,0,0,0,0,0,0, 1,1,1,1,1,1,1,1, 2,2,2,2,2,2,2,2, 3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,
4,4,4,4,4,4,4,4, 5,5,5,5,5,5,5,5, 6,6,6,6,6,6,6,6, 7,7,7,7, 8,8];
receiver = [1,1,2,2,3,3,3,3, 2,2,3,3,4,4,4,4, 3,3,4,4,5,5,5,5, 4,4,5,5,6,6,6,6,8,8,8,8,8,8,
5,5,6,6,7,7,7,7, 6,6,7,7,8,8,8,8, 7,7,8,8,9,9,9,9, 8,8,9,9, 9,9];
d = zeros(68,1);
```

Main calculation

```
for i = 1:length(d)
    switch i
        case {1,9,17,25,39,47,55,63,67}
            pos1 = find([Tomo.source] == source(i) & [Tomo.receiver] == receiver(i));
            d(i) = round((Tomo(pos1).ver_length / 2) / sin(Tomo(pos1).angle), 5);
        case {3,11,19,27,41,49,57,65}
            pos1 = find([geometry.sensor] == source(i));
            pos2 = find([geometry.sensor] == receiver(i-1));
            d(i) = round(geometry(pos2).distance_from_edge -
geometry(pos1).distance_from_edge, 5);
        case {2,4,10,12,18,20,26,28,40,42,48,50,56,58,64,66,68}
            pos1 = find([Tomo.source] == source(i) & [Tomo.receiver] == receiver(i));
            d(i) = round(Tomo(pos1).SR_length- d(i-1), 5);
        case {5,13,21,29,33,43,51,59}
            pos1 = find([geometry.sensor] == source(i));
            pos2 = pos1 + 1;
            pos3 = find([Tomo.source] == source(i) & [Tomo.receiver] == receiver(i));
            d(i) = round((geometry(pos2).distance_from_edge -
geometry(pos1).distance_from_edge) / cos(Tomo(pos3).angle), 5);
        case {6,14,22,30,44,52,60}
            pos1 = find([Tomo.source] == source(i) & [Tomo.receiver] == receiver(i));
            S_length = (Tomo(pos1).ver_length / 2) / sin(Tomo(pos1).angle);
            d(i) = S_length - d(i-1);
        case{8,16,24,32,38,46,54,62}
            pos1 = find([geometry.sensor] == receiver(i));
            pos2 = pos1 - 1;
            pos3 = find([Tomo.source] == source(i) & [Tomo.receiver] == receiver(i));
            d(i) = round((geometry(pos1).distance_from_edge -
geometry(pos2).distance_from_edge) / cos(Tomo(pos3).angle), 5);
        case 34
            pos1 = find([geometry.sensor] == source(i));
            pos2 = pos1 + 2;
            pos3 = find([Tomo.source] == source(i) & [Tomo.receiver] == receiver(i));
            d(i) = round((geometry(pos2).distance_from_edge -
geometry(pos1).distance_from_edge) / cos(Tomo(pos3).angle) - d(i-1), 5);
        case 35
            pos1 = find([Tomo.source] == source(i) & [Tomo.receiver] == receiver(i));
            S_length = (Tomo(pos1).ver_length / 2) / sin(Tomo(pos1).angle);
            d(i) = S_length - d(i-1) - d(i-2);
    end
end

for i = 1:length(d)
    switch i
        case{7,15,23,31,45,53,61}
            pos1 = find([Tomo.source] == source(i) & [Tomo.receiver] == receiver(i));
            S_length = (Tomo(pos1).ver_length / 2) / sin(Tomo(pos1).angle);
            d(i) = S_length - d(i+1);
        case 37
            pos1 = find([geometry.sensor] == receiver(i));
            pos2 = pos1 - 2;
            pos3 = find([Tomo.source] == source(i) & [Tomo.receiver] == receiver(i));
            d(i) = round((geometry(pos1).distance_from_edge -
geometry(pos2).distance_from_edge) / cos(Tomo(pos3).angle) - d(i+1), 5);
    end
end
```

```
for i = 1:length(d)
    switch i
        case 36
            pos1 = find([Tomo.source] == source(i) & [Tomo.receiver] == receiver(i));
            S_length = (Tomo(pos1).ver_length / 2) / sin(Tomo(pos1).angle);
            d(i) = S_length - d(i+1) - d(i+2);
    end
end
i=54;

% Save the structure to a .mat file
save('Distance.mat', 'd');
```

Published with MATLAB® R2022b

Appendix III.4.1.4 – Coor_cell.m

VNPonson Tomography Graphs - V2.0

initial Tomography Code p4v5 written by Vincent Ponson (v.n.ponson@student.tudelft.nl)
Setting up the Struct file - the coordinates for the vel. fields & dt lines (UPV&CWI)

```
% Files needed:  TMD_(CWI/UPV), Geometry
% Files created: coor_cells
```

Setting up Indexes and Preallocate memory

Source - Cell Pairings

```
cell_x1 = [0,0,1,1,2,2,3,3,4,4,5,5,6,6,7,7,8,8];
cell_x2 = [1,1,2,2,3,3,4,4,5,5,6,6,7,7,8,8,9,9];
cell_y1 = zeros(1,18);

for i = 1:18
    cell_coor(i).cell = i;

    pos1 = find([geometry.sensor]==cell_x1(i));
    pos2 = find([geometry.sensor]==cell_x2(i));
    pos3 = find([geometry.sensor]==cell_y1(i));

    cell_coor(i).left_bottom_x = geometry(pos1).distance_from_edge;
    cell_coor(i).right_top_x   = geometry(pos2).distance_from_edge;

    switch i
        case {1,3,5,7,9,11,13,15,17}
            cell_coor(i).left_bottom_y = geometry(pos3).height + 75/1000;
            cell_coor(i).right_top_y   = geometry(pos3).height + 150/1000;
        case {2,4,6,8,10,12,14,16,18}
            cell_coor(i).left_bottom_y = geometry(pos3).height;
            cell_coor(i).right_top_y   = geometry(pos3).height + 75/1000;
    end
end

% Source - dt-Line Pairings
line_x1 = [1,2,3,4,5,6,7,8];
line_x2 = [2,3,4,5,6,7,8,9];
```

```
line_y = zeros(1,7);

for i = 1:7
    line_coor(i).line = i;

    pos1 = find([geometry.sensor]==line_x1(i));
    pos2 = find([geometry.sensor]==line_x2(i));
    pos3 = find([geometry.sensor]==line_y(i));

    line_coor(i).left_x  = geometry(pos1).distance_from_edge;
    line_coor(i).right_x = geometry(pos2).distance_from_edge;

    line_coor(i).left_y  = geometry(pos3).height +70/1000;
    line_coor(i).right_y = geometry(pos3).height + 80/1000;
end

% Save the structure to a .mat file
save('coor_cell.mat', 'cell_coor');
save('coor_line.mat', 'line_coor');
```

Published with MATLAB® R2022b

Appendix III.4.1.5 – Initial_Vel_calc.m

VNPonson Tomography Graphs - V2.0

initial Tomography Code p5v5 written by Vincent Ponson (v.n.ponson@student.tudelft.nl)
Setting up the Struct file - the initial velocity of each cell (UPV&CWI)

```
% Files needed:  TMD_(UPV or CWI)
% Files created: initial Velocity: (Vel_in) & (VEL_ine)
```

Setting up Indexes and Preallocate memory

Source - Receiver Distance Pairing

```
source   = [0,0,0, 1,1,1,1, 2,2,2,2,2, 3,3,3,3,3,3,3, 4,4,4,4,4,4, 5,5,5,5,5,5, 6,6,6,6,6,6,
7,7,7,7,7, 8,8,8,8,8, 9,9,9];
receiver = [1,2,3, 0,2,3,4, 0,1,3,4,5, 0,1,2,4,5,6,8, 1,2,3,5,6,7, 2,3,4,6,7,8, 3,4,5,7,8,9,
4,5,6,8,9, 3,5,6,7,9, 6,7,8];

% Indexes to extract specific Pairs
Pair_indices = [1,5,10,16,23,29,35,41,46];

% Preset loadstep count, CHANGE
loadsteps_UPV = 28; % UPV calc
loadsteps_CWI = 13; % CWI calc

% Preallocate memory for 'v' and 'Vel_in' structures
v = zeros(length(source), loadsteps_UPV);
Vel_in = struct();
Vel_ine = struct();

% Preallocate memory for adjuested EPS list
eps = zeros(length(source), loadsteps_CWI);
EPS_in = struct();
```

UPV Velocity calculations

```
for i = 1:loadsteps_UPV
    for j = 1:length(source)
        pos1 = find([Tomo.source] == source(j) & [Tomo.receiver] == receiver(j));
        pos2 = find([Tomo.source] == receiver(j) & [Tomo.receiver] == source(j));
        v1 = Tomo(pos1).velocity;
        v2 = Tomo(pos2).velocity;

        v(j, i) = (v1(i) + v2(i)) / 2;
    end
end

% Assign values from selected velocities indices to velocity structures
for k = 1:length(Pair_indices)
    Vel_in.(['v', num2str(k)]) = v(Pair_indices(k), :); %UPV, initial velocity
    Vel_ine.(['v', num2str(k)]) = [v(Pair_indices(k), 1), zeros(1,(loadsteps_CWI-1))];
end
```

Adjusted EPS list

```
for i = 1:loadsteps_CWI
    for j = 1:length(Pair_indices)
        pos1 = find([Tomo.source] == source(j) & [Tomo.receiver] == receiver(j));
        pos2 = find([Tomo.source] == receiver(j) & [Tomo.receiver] == source(j));
        eps1 = Tomo(pos1).EPS;
        eps2 = Tomo(pos2).EPS;

        eps(j, i) = (eps1(i) + eps2(i)) / 2;
    end
end

% Assign values from selected EPS indices to EPS structure
for k = 1:length(Pair_indices)
    EPS_in.(['eps', num2str(k)]) = eps(Pair_indices(k), :);
end
```

CWI Velocity calculations

```
for i=1:length(Pair_indices)
    prod_eps = 1; % Reset cumulative product for each 'v'
    for j=2:loadsteps_CWI
            current_eps = EPS_in.(sprintf('eps%d', i))(j);
            prod_eps = prod_eps * (current_eps + 1); % Cumulative product

            Vel_ine.(sprintf('v%d', i))(j) = Vel_ine.(sprintf('v%d', i))(1) * prod_eps; %CWI
initial time
    end
end
```

Published with MATLAB® R2022b

written by Vincent Ponson (v.n.ponson@student.tudelft.nl) Solves for Velocity and Interface Time interferance, using FMINCON (CWI)

```
% Files needed:  In_Velocity(_eps), EPS, Time_eps, Distance
% Files created: Interface Timeinterfereance(dt_time_calc), velocity in Cells
(velocity_calc), EPS in Cells (EPS_sol)
```

Define Loadsteps and Arrays

```
loadsteps_CWI = 13;
unk       = 25;
unk_vel   = 18;
unk_time  = 7;

% Preallocate memory for each cell's EPS
eps_sol = zeros(loadsteps_CWI, unk_vel);
EPS_cell = struct();

for i = 1:9
    av(2*i-1).velocity = Vel_ine.(['v' num2str(i)]);
    av(2*i).velocity = Vel_ine.(['v' num2str(i)]);
end
```

Solving the SoE, using Fmincon

Main Calcualtion

```
tic;
for i = 1:loadsteps_CWI
    % Initial guess
    initial_guess = zeros(1,25);

    for j=1:unk_vel
        initial_guess(j) = av(j).velocity(i);
    end

        % Lower bounds; dt > 0. Upper bounds; dt < 1e-4
    lb = [-inf(1, 18), zeros(1, 7)];
    ub = [inf(1, 18), 1e-4*ones(1,7)];

    % Weights
    %weights = [1, 1, 1, 1, 1, 1, 5, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 5, 5, 5, 5, 5, 5, 5];
    %help    = [1, 2, 3, 4, 5, 6, 7, 8, 9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25]

    % Define the objective function & nonlinear contraint (F(x)=0) for fmincon
    objective = @(x) sum(system_of_equations(x, d, te, i).^2);
    %nonlcon = @(x) deal([], system_of_equations(x, d, t, i));

    % Optimisations options
    options = optimoptions('fmincon', 'Display', 'iter', 'TolFun', 1e-10);

    [solution, fval] = fmincon(objective, initial_guess, [], [], [], [], lb, ub, [],
options);

    % Store the solution for velocities and dt variables
    sol(i,:)             = solution;
```

```
    velocities_sol(i, :) = solution(1:18);
    dt_sol(i, :)         = solution(19:25);
end
toc;

% Store results into the velocity and IFTime structs
for k = 1:unk_vel
    velocity(k).cell = k;
    velocity(k).velocity = velocities_sol(:, k);
end

for k = 1:unk_time
    IFTime(k).dt_number = k;
    IFTime(k).time = dt_sol(:, k);
end
```

Back-calculate EPS

```
for j = 1:unk_vel
    velocity_vals = velocities_sol(:, j);
    Vel_0 = velocity_vals(1);

    % Reset cumulative product for each loadstep
    prod_eps = 1;

    for i = 2:loadsteps_CWI
        % Compute eps using the modified product formula
        eps_sol(i, j) = (velocity_vals(i) / (prod_eps * Vel_0)) - 1;

        % Update cumulative product for the next iteration
        prod_eps = prod_eps * (eps_sol(i, j) + 1);
    end
end

% Store results into the eps struct
for k = 1:unk_vel
    EPS_cell(k).cell = k;
    EPS_cell(k).eps = eps_sol(:, k);
end
```

Function defining the System of Equations

SoE: 25 Equations to 25 Unknowns

```
function F = system_of_equations(x, d, te, i)
    % Extract variables from x
    v1 = x(1); v2 = x(2); v3 = x(3); v4 = x(4); v5 = x(5);
    v6 = x(6); v7 = x(7); v8 = x(8); v9 = x(9); v10 = x(10);
    v11 = x(11); v12 = x(12); v13 = x(13); v14 = x(14); v15 = x(15);
    v16 = x(16); v17 = x(17); v18 = x(18);
    dt1 = x(19); dt2 = x(20); dt3 = x(21);
    dt4 = x(22); dt5 = x(23); dt6 = x(24); dt7 = x(25);

    % Define the equations
    F(1)  = d(1)/v2  + d(2)/v1  - te.t1(i);
    F(2)  = d(3)/v2  + d(4)/v4  - te.t2(i);
    F(3)  = d(5)/v2  + d(6)/v4  + d(7)/v3  + d(8)/v5  + dt1 - te.t3(i);
```

```
    F(4)  = d(9)/v3  + d(10)/v4 + dt1 - te.t4(i);
    F(5)  = d(11)/v3 + d(12)/v5 - te.t5(i);
    F(6)  = d(13)/v3 + d(14)/v5 + d(15)/v6 + d(16)/v8 + dt2 - te.t6(i);

    F(7)  = d(17)/v6 + d(18)/v5 + dt2 - te.t7(i);
    F(8)  = d(19)/v6 + d(20)/v8 - te.t8(i);
    F(9)  = d(21)/v6 + d(22)/v8 + d(23)/v7 + d(24)/v9 + dt3 - te.t9(i);

    F(10) = d(25)/v7 + d(26)/v8 + dt3  - te.t10(i);
    F(11) = d(27)/v7 + d(28)/v9 - te.t11(i);
    F(12) = d(29)/v7 + d(30)/v9 + d(31)/v10 + d(32)/v12 + dt4 - te.t12(i);
    F(13) = d(33)/v7 + d(34)/v9 + d(35)/v11 + d(36)/v12 + d(37)/v12 + d(38)/v16 + dt5 -
te.t13(i);

    F(14) = d(39)/v10 + d(40)/v9 + dt4 - te.t14(i);
    F(15) = d(41)/v10 + d(42)/v12 - te.t15(i);
    F(16) = d(43)/v10 + d(44)/v12 + d(45)/v11 + d(46)/v13 + dt5 - te.t16(i);

    F(17) = d(47)/v11 + d(48)/v12 + dt5  - te.t17(i);
    F(18) = d(49)/v11 + d(50)/v13 - te.t18(i);
    F(19) = d(51)/v11 + d(52)/v13 + d(53)/v14 + d(54)/v16 + dt6 - te.t19(i);

    F(20) = d(55)/v14 + d(56)/v13 + dt6 - te.t20(i);
    F(21) = d(57)/v14 + d(58)/v16 - te.t21(i);
    F(22) = d(59)/v14 + d(60)/v16 + d(61)/v15 + d(62)/v17 + dt7 - te.t22(i);

    F(23) = d(63)/v15 + d(64)/v16 + dt7 - te.t23(i);
    F(24) = d(65)/v15 + d(66)/v17 - te.t24(i);

    F(25) = d(67)/v18 + d(68)/v17 - te.t25(i);
end
```

Published with MATLAB® R2022b

Appendix III.4.3 – Accuracy_calc.m

VNPonson Tomography Accuracy Calc - V1.5

Main Accuracy Calculation written by Vincent Ponson (v.n.ponson@student.tudelft.nl)
Calculates the several accuracy variables RMSD & ID, (UPV/CWI)

```
% Files needed: Distance, Time, sol (matrix)
% Files created: Accuracy struc
```

Loadsteps & Rearinging Time struc

```
loadsteps = 13; % Change
equations = 25;

for i=1:loadsteps
    t_ac(i).loadstep = i;
    for j = 1:equations
        t_ac(i).Time(j) = te.(['t' num2str(j)])(i);
    end
end
```

Main Calc

```
for i=1:loadsteps
    t_acc  = t_ac(i).Time;
    t_calc = system_of_equations(sol(i,:), d);
    diff   = t_acc - t_calc;

    % Calculate RMSD
    rmsd_values(i) = sqrt(mean(diff.^2));

    % Store results into the RMSD struct
    accuracy(i).loadstep   = i;
    accuracy(i).time_meas  = t_acc;
    accuracy(i).time_calc  = t_calc;
    accuracy(i).difference = diff;

    for j = 1:equations
        accuracy(i).ID(j) = (accuracy(i).time_calc(j) - t_ac(i).Time(j)) /
t_ac(i).Time(j)*100;
    end

    accuracy(i).RMSD       = rmsd_values(i);
end
```

Function defining the System of Equations

SoE: 25 Equations to 25 Unknowns with out T

```
function F = system_of_equations(x, d)
    v1 = x(1); v2 = x(2); v3 = x(3); v4 = x(4); v5 = x(5);
    v6 = x(6); v7 = x(7); v8 = x(8); v9 = x(9); v10 = x(10);
    v11 = x(11); v12 = x(12); v13 = x(13); v14 = x(14); v15 = x(15);
    v16 = x(16); v17 = x(17); v18 = x(18);
    dt1 = x(19); dt2 = x(20); dt3 = x(21);
    dt4 = x(22); dt5 = x(23); dt6 = x(24); dt7 = x(25);

    % Define the equations
    F(1)  = d(1)/v2  + d(2)/v1;
    F(2)  = d(3)/v2  + d(4)/v4;
    F(3)  = d(5)/v2  + d(6)/v4  + d(7)/v3  + d(8)/v5  + dt1;

    F(4)  = d(9)/v3  + d(10)/v4 + dt1;
    F(5)  = d(11)/v3 + d(12)/v5;
    F(6)  = d(13)/v3 + d(14)/v5 + d(15)/v6 + d(16)/v8 + dt2;

    F(7)  = d(17)/v6 + d(18)/v5 + dt2;
    F(8)  = d(19)/v6 + d(20)/v8;
    F(9)  = d(21)/v6 + d(22)/v8 + d(23)/v7 + d(24)/v9 + dt3;

    F(10) = d(25)/v7 + d(26)/v8 + dt3;
    F(11) = d(27)/v7 + d(28)/v9;
    F(12) = d(29)/v7 + d(30)/v9 + d(31)/v10 + d(32)/v12 + dt4;
    F(13) = d(33)/v7 + d(34)/v9 + d(35)/v11 + d(36)/v12 + d(37)/v12 + d(38)/v16 + dt5;

    F(14) = d(39)/v10 + d(40)/v9 + dt4;
    F(15) = d(41)/v10 + d(42)/v12;
    F(16) = d(43)/v10 + d(44)/v12 + d(45)/v11 + d(46)/v13 + dt5 ;
```

```
    F(17) = d(47)/v11 + d(48)/v12 + dt5;
    F(18) = d(49)/v11 + d(50)/v13;
    F(19) = d(51)/v11 + d(52)/v13 + d(53)/v14 + d(54)/v16 + dt6;

    F(20) = d(55)/v14 + d(56)/v13 + dt6;
    F(21) = d(57)/v14 + d(58)/v16;
    F(22) = d(59)/v14 + d(60)/v16 + d(61)/v15 + d(62)/v17 + dt7 ;

    F(23) = d(63)/v15 + d(64)/v16 + dt7;
    F(24) = d(65)/v15 + d(66)/v17;

    F(25) = d(67)/v18 + d(68)/v17;
end
```

Published with MATLAB® R2022b

Appendix III.4.4 – Accuracy Plots

Appendix III.4.4.1 – Plot_ID.m

VNPonson Tomography Accuracy Plotter - V2.0
written by Vincent Ponson (v.n.ponson@student.tudelft.nl) Plots ID value

```
% Files needed: Accuracy struc
% Files created: ID Plots
```

Main Code

```
loadsteps = 13; %CHANGE

for i = 1:loadsteps
    ID = accuracy(i).ID;
    fig = simplePlot(ID,i);

    filename = sprintf('S10H1D_VEL_TOMO_ID_%d.jpg', (i - 1) * 50); %Change .jpg or .svg
    saveas(fig, filename);
    close(fig);
end
```

Plot Function

```
function fig = simplePlot(x, i)
    fig = figure;
    fig.Position = [50, 50, 2000, 1000];
    stem(1:length(x), x, 'b', 'filled', 'lineWidth', 3.0, 'MarkerSize', 5.0);

    yline(0, 'k'); % line at y = 0

    % Add title and labels
    xlabel('Equations [-]', 'FontSize', 16.0);
    ylabel('Individual Difference [%]', 'FontSize', 16.0);
    title(sprintf('Beam S10H1D, UPV-velocity Model at loadstep %d [kN]', (i - 1) * 50),
'FontSize', 20.0); % CHANGE
end
```

END
Published with MATLAB® R2022b

Appendix III.4.4.2 – Plot_RSMD.m

VNPonson Tomography Accuracy Plotter - V2.0
written by Vincent Ponson (v.n.ponson@student.tudelft.nl) Plots RSMD value

```
% Files needed: Accuracy struc
% Files created: RMSD Plot
```

Main Code

```
x = [accuracy.RMSD];
%y = [accuracy30.RMSD]; % Add when comapring
fig = simplePlot(x); % Add when comapring

filename = sprintf('S10H1D_VEL_TOMO_RMSD.jpg'); %Change .jpg or .svg
saveas(fig, filename);
close(fig);
```

Plot Function

```
function fig = simplePlot(x)
    fig = figure;
    fig.Position = [50, 50, 2000, 1000];
    plot(1:length(x), x, 'b-', 'LineWidth', 1.5);
    %plot(1:length(y), x, 'r-', 'LineWidth', 1.5); ) % Add when comapring

    yline(0, 'k');

    % Add title and labels
    xlabel('Loadsteps [-]', 'FontSize', 16.0);
    ylabel('RMSD [s]', 'FontSize', 16.0);
    title('Beam S10H1D, RMSD-value', 'FontSize', 20.0); % CHANGE
    %legend('Model '); % Add when comapring
end
```

END
Published with MATLAB® R2022b

Appendix III.5 – Ray Path Plotting Code

Appendix III.5.1 – plot_figure.m

Note: Depending on the parameter to be plotted, only the relevant section of the code should be run.

v1.0 by Hao Cheng, 11/01/2023, email: h.cheng-2@tudelft.nl
v2.4 adjusted by Vincent Ponson, 22/11/2025, email: v.ponson@student.tudelft.nl velocity change visualization

load files

sensor location in the girder
plot wave trajectory between sensors

```
% Define window and ratio of the saved figure
zoom_x = [0.0 3.5];
zoom_y = [0.75 1.07];

width  = 2000;
height = width * (1/5);

dx = 0.015;
dy = 0.015;

% UPV
loadsteps = 28;

for i=1:90
    velocity(i).velocity=velocity(i).velocity./velocity(i).velocity(1);
end

for load_number = 1:loadsteps
    open('empty_figure_UPV_zoom.fig');
    fig = gcf;

    % Set figure size with ratio
    set(fig, 'Units','pixels');
    set(fig, 'Position',[0 0 width height]);

    hold on;
    [color1, mean_CC1] = color_plot(0, [1;2;3], load_number, geometry, velocity);
    [color2, mean_CC2] = color_plot(1, [2;3;4], load_number, geometry, velocity);
    [color3, mean_CC3] = color_plot(2, [3;4;5], load_number, geometry, velocity);
    [color4, mean_CC4] = color_plot(3, [4;5;6], load_number, geometry, velocity);
    [color5, mean_CC5] = color_plot(4, [5;6;7], load_number, geometry, velocity);
    [color6, mean_CC6] = color_plot(5, [6;7;8], load_number, geometry, velocity);
    [color7, mean_CC7] = color_plot(6, [7;8;9], load_number, geometry, velocity);
    [color8, mean_CC8] = color_plot(7, [8;9],   load_number, geometry, velocity);
    [color9, mean_CC9] = color_plot(8, [9],     load_number, geometry, velocity);
    hold off;

    % Apply window for saved plot
    xlim(zoom_x)
    ylim(zoom_y)

    % SA naming
    for sa = 0:9
        x = geometry(sa+1).distance_from_edge;
        y = geometry(sa+1).height;
```

```
        if mod(sa,2) == 0
            x_txt = x - dx;
            y_txt = y - dy;
            hAlign = 'right';
            vAlign = 'top';
        else
            x_txt = x - dx;
            y_txt = y + dy;
            hAlign = 'right';
            vAlign = 'bottom';
        end

        text(x_txt, y_txt, sprintf('SA%d', sa), 'FontSize', 14, ...
            'FontWeight', 'bold', 'HorizontalAlignment', hAlign, ...
            'VerticalAlignment', vAlign, 'Color', [0 0 1]);
    end

    filename = sprintf('S10H1D_Vel_%d.jpg', (load_number - 1) * 50);
    saveas(fig, filename);

    close(fig);
end

% CC
loadsteps = 15;

for load_number = 1:loadsteps
    openfig('empty_figure_CC_zoom.fig');
    fig = gcf;

    % Set figure size with ratio
    set(fig, 'Units','pixels');
    set(fig, 'Position',[0 0 width height]);

    hold on;
    [color1, mean_CC1] = color_plot(0, [1;2;3], load_number, geometry, DWI);
    [color2, mean_CC2] = color_plot(1, [2;3;4], load_number, geometry, DWI);
    [color3, mean_CC3] = color_plot(2, [3;4;5], load_number, geometry, DWI);
    [color4, mean_CC4] = color_plot(3, [4;5;6], load_number, geometry, DWI);
    [color5, mean_CC5] = color_plot(4, [5;6;7], load_number, geometry, DWI);
    [color6, mean_CC6] = color_plot(5, [6;7;8], load_number, geometry, DWI);
    [color7, mean_CC7] = color_plot(6, [7;8;9], load_number, geometry, DWI);
    [color8, mean_CC8] = color_plot(7, [8;9],   load_number, geometry, DWI);
    [color9, mean_CC9] = color_plot(8, [9],     load_number, geometry, DWI);
    hold off;

    % Apply window for saved plot
    xlim(zoom_x)
    ylim(zoom_y)
```

```
    % SA naming
    for sa = 0:9
        x = geometry(sa+1).distance_from_edge;
        y = geometry(sa+1).height;

        if mod(sa,2) == 0
            x_txt = x - dx;
            y_txt = y - dy;
            hAlign = 'right';
            vAlign = 'top';
        else
            x_txt = x - dx;
            y_txt = y + dy;
            hAlign = 'right';
            vAlign = 'bottom';
        end

        text(x_txt, y_txt, sprintf('SA%d', sa), 'FontSize', 14, ...
            'FontWeight', 'bold', 'HorizontalAlignment', hAlign, ...
            'VerticalAlignment', vAlign, 'Color', [0 0 1]);
    end


    filename = sprintf('S10H1D_CC_%d.jpg', (load_number) * 50); %BOTH .svg as .jpg
    saveas(fig, filename);

    close(fig); % Close the figure to avoid memory issues
end

% EPS
loadsteps = 15;

for load_number = 1:loadsteps
    openfig('empty_figure_EPS_zoom.fig');
    fig = gcf;

    % Set figure size with ratio
    set(fig, 'Units','pixels');
    set(fig, 'Position',[0 0 width height]);

    hold on;
    [color1, mean_CC1] = color_plot(0, [1;2;3], load_number, geometry, DWI);
    [color2, mean_CC2] = color_plot(1, [2;3;4], load_number, geometry, DWI);
    [color3, mean_CC3] = color_plot(2, [3;4;5], load_number, geometry, DWI);
    [color4, mean_CC4] = color_plot(3, [4;5;6], load_number, geometry, DWI);
    [color5, mean_CC5] = color_plot(4, [5;6;7], load_number, geometry, DWI);
    [color6, mean_CC6] = color_plot(5, [6;7;8], load_number, geometry, DWI);
    [color7, mean_CC7] = color_plot(6, [7;8;9], load_number, geometry, DWI);
```

```
    [color8, mean_CC8] = color_plot(7, [8;9],   load_number, geometry, DWI);
    [color9, mean_CC9] = color_plot(8, [9],     load_number, geometry, DWI);
    hold off;

    % Apply window for saved plot
    xlim(zoom_x)
    ylim(zoom_y)

    % SA naming
    for sa = 0:9
        x = geometry(sa+1).distance_from_edge;
        y = geometry(sa+1).height;

        if mod(sa,2) == 0
            x_txt = x - dx;
            y_txt = y - dy;
            hAlign = 'right';
            vAlign = 'top';
        else
            x_txt = x - dx;
            y_txt = y + dy;
            hAlign = 'right';
            vAlign = 'bottom';
        end

        text(x_txt, y_txt, sprintf('SA%d', sa), 'FontSize', 14, ...
            'FontWeight', 'bold', 'HorizontalAlignment', hAlign, ...
            'VerticalAlignment', vAlign, 'Color', [0 0 1]);
    end


    filename = sprintf('S10H1D_EPS_%d.jpg', (load_number) * 50); %BOTH .svg as .jpg
    saveas(fig, filename);

    close(fig); % Close the figure to avoid memory issues
end
```

range of color bar

```
colorbar('Ticks',[0,0.2,0.4,0.6,0.8,1],'TickLabels',{'0.75','0.97','0.99','1.01','1.03','1'},'LineWidth',1.5);
```

END

Published with MATLAB® R2022b

Appendix III.5.2 – color_plot.m

Note: Depending on which parameter is being plotted, the other two must be disabled by commenting them out using %, as shown in the example code.

```
function
[color,mean_CC_output,mean_velocity_output]=color_plot(source,receiver,load_level_number,geom
etry,velocity)
```

v1.0 by Hao Cheng, 11/01/2023, email: h.cheng-2@tudelft.nl
v1.2 by Vincent Ponson 06/08/2024, email: v.n.ponson@student.tudelft.nl plotting with selected color

CODE
Find positions of the source and receivers

```
pos_location_source=find([geometry.sensor]==source);

% Initial receiver positions array
pos_location_receiver = zeros(length(receiver), 1);

for i=1:length(receiver)
    pos_location_receiver(i,:)=find([geometry.sensor]==receiver(i));
end

% Initial color array
color = ones(length(receiver), 3);

% UPV - Iterate over receivers to calculate mean_velocity and assign colors
% for i = 1:length(receiver)
%     pos1 = find([velocity.source] == source & [velocity.receiver] == receiver(i));
%     pos2 = find([velocity.source] == receiver(i) & [velocity.receiver] == source);
%
%     mean_CC_output=0;
%
mean_velocity=(velocity(pos1).velocity(load_level_number)+velocity(pos2).velocity(load_level_
number))/2;
%     if mean_velocity>1  % range of color bar, can be set as the input (upepr)
%         mean_velocity=1;  % range of color bar, can be set as the input (upper)
%     end
%
%     if mean_velocity<0.8 % range of color bar, can be set as the input (lower)
%         mean_velocity=0.8; % range of color bar, can be set as the input (lower)
%     end
%     color(i,:)=color_find_VEL(0.8,1,mean_velocity); % range of color bar, can be set as the
input (lower, upper)
%  end

% EPS - Iterate over receivers to calculate mean_velocity, mean_CC, and assign colors
% for i = 1:length(receiver)
%     pos1 = find([velocity.source] == source & [velocity.receiver] == receiver(i));
%     pos2 = find([velocity.source] == receiver(i) & [velocity.receiver] == source);
%
%     white = [1, 1, 1];
%
%     mean_velocity = (velocity(pos1).EPS(load_level_number) +
velocity(pos2).EPS(load_level_number)) / 2;
%     mean_CC = (velocity(pos1).CC(load_level_number) + velocity(pos2).CC(load_level_number))
/ 2;
%     mean_velocity_output(i) = mean_velocity;
```

```
%     mean_CC_output(i) = mean_CC;
%
%     % Check and adjust the mean_velocity to fit within the color range
%     if mean_velocity > 0  % range of color bar, can be set as the input (upper)
%         mean_velocity = 0;  % range of color bar, can be set as the input (upper)
%     end
%
%     % Assign color based on mean_velocity
%     color(i, :) = color_find_EPS(-2e-3, 0, mean_velocity);  % range of color bar, can be
set as the input (lower, upper)
%
%     % Assign white color for mean_CC values < 0.8
%     if mean_CC_output(i) < 0.8
%         color(i, :) = white;
%     end
% end

% % CWI - Iterate over receivers to calculate mean_CC, and assign colors
for i = 1:length(receiver)
    pos1 = find([velocity.source] == source & [velocity.receiver] == receiver(i));
    pos2 = find([velocity.source] == receiver(i) & [velocity.receiver] == source);

    mean_velocity = (velocity(pos1).EPS(load_level_number) +
velocity(pos2).EPS(load_level_number)) / 2;
    mean_CC = (velocity(pos1).CC(load_level_number) + velocity(pos2).CC(load_level_number)) /
2;
    mean_velocity_output(i) = mean_velocity;
    mean_CC_output(i) = mean_CC;

    % Check and adjust the mean_velocity to fit within the color range
    if mean_CC > 1  % range of color bar, can be set as the input (upper)
        mean_CC = 1;  % range of color bar, can be set as the input (upper)
    end

    color(i, :) = color_find_CC(0.8, 1, mean_CC);  % range of color bar, can be set as the
input (lower, upper)
end

% Plot lines based on the calculated colors
for i=1:length(receiver)
    line([geometry(pos_location_source).distance_from_edge,...
        geometry(pos_location_receiver(i)).distance_from_edge],...
        [geometry(pos_location_source).height,...
        geometry(pos_location_receiver(i)).height],...
        'color',color(i,:),'linewidth',4);
end
```

END
Published with MATLAB® R2022b

Appendix III.5.3 – Code to determine the Colour

Appendix III.5.3.1 – color_find_VEL.m

```
function [color]=color_find(bottom,top,velocity)
```

v1.0 by Hao Cheng, 11/01/2023, email: h.cheng-2@tudelft.nl
v1.1 adjusted by Vincent Ponson, 20/07/2024, email: v.ponson@student.tudelft.nl find the color based on the selected range red to green: typical colors used for numerical simulation

CODE

```
delta=(top-bottom)/2;
if velocity<bottom+delta
    a=(velocity-bottom)/delta;
    color=[1,a,0];
end
if velocity>=bottom+delta
    a=1-((velocity-bottom-delta)/delta);
    color=[a,1,0];
end

end
```

END
Published with MATLAB® R2022b

Appendix III.5.3.2 – color_find_CC.m

```
function [color]=color_find(bottom,top,velocity)
```

v1.0 by Hao Cheng, 11/01/2023, email: h.cheng-2@tudelft.nl
find the color based on the selected range red to green: typical colors used for numerical simulation

CODE
Ensure velocity is within the bounds

```
if velocity < bottom
    velocity = bottom;
elseif velocity > top
    velocity = top;
end

delta=(top-bottom)/2;

if velocity<bottom+delta
    a=(velocity-bottom)/delta;
    color=[1,a,0];% Transition from red to yellow
end
if velocity>=bottom+delta
    a=1-((velocity-bottom-delta)/delta);
    color=[a,1,0];% Transition from yellow to green
end

end
```

END
Published with MATLAB® R2022b

Appendix III.5.3.3 – color_find_EPS.m

```
function [color]=color_find(bottom,top,velocity)
```

v1.0 by Hao Cheng, 11/01/2023, email: h.cheng-2@tudelft.nl
find the color based on the selected range red to green: typical colors used for numerical simulation

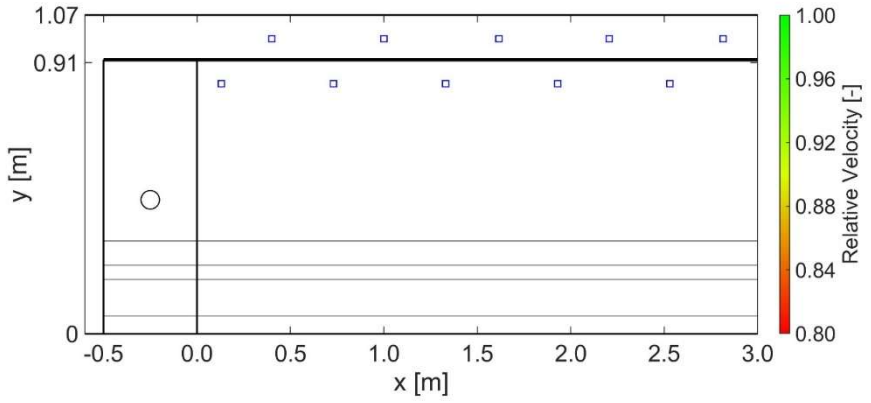
CODE
Ensure velocity is within the bounds

```
if velocity < bottom
    velocity = bottom;
elseif velocity > top
    velocity = top;
end

delta = (top - bottom) / 4;

if velocity < bottom + delta
    t = (velocity - bottom) / delta;
    color = [0, 0, 1] * (1 - t) + [0.5, 0, 0.5] * t;% Transition from blue to purple
elseif velocity < bottom + 2 * delta
    t = (velocity - (bottom + delta)) / delta;
    color = [0.5, 0, 0.5] * (1 - t) + [1, 0, 0] * t;% Transition from purple to red
elseif velocity < bottom + 3 * delta
    t = (velocity - (bottom + 2 * delta)) / delta;
    color = [1, 0, 0] * (1 - t) + [1, 1, 0] * t;% Transition from red to yellow
else
    t = (velocity - (bottom + 3 * delta)) / delta;
    color = [1, 1, 0] * (1 - t) + [0, 1, 0] * t;% Transition from yellow to green
end

end
```

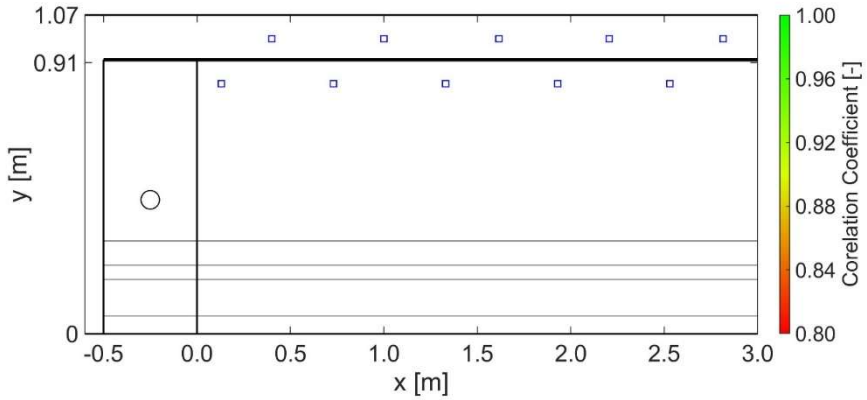
END
Published with MATLAB® R2022b

Appendix III.5.4 – Empty Figure

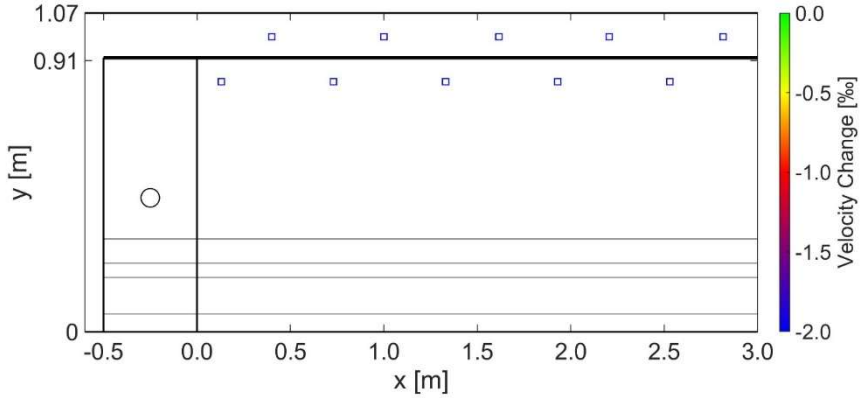
Appendix III.5.4.1 – empty_figure_UPV.fig



Appendix III.5.4.2 – empty_figure_CC.fig



Appendix III.5.4.3 – empty_figure_EPS.fig



Appendix III.6 – Tomography Plotting Code

Appendix III.6.1 – Figure Plotter

Appendix III.6.1.1 – plot_figure.m

Note: Only used for the UPV calculations.

Tomography Plotter - V2.0
v2.0 by Vincent Ponson (v.n.ponson@student.tudelft.nl) velocity change visualization - Tomography

load files
sensor location in the girder
plot wave trajectory between sensors

Define window and ratio of the saved figure

```
zoom_x = [0.0 3.5];
zoom_y = [0.75 1.07];

width  = 2000;
```

```
height = width * (1/5);

dx = 0.015;
dy = 0.015;

% UPV_Tomo
loadsteps = 28; %CHANGE

% Cells & dt-Lines
for i=1:loadsteps
    velocity(i).velocity=velocity(i).velocity./velocity(i).velocity(1);
end

for i=1:7
    IFTime(i).time=IFTime(i).time/IFTime(i).time(1);
end

for load_number = 1:loadsteps
    open('empty_figure_UPV_zoom.fig');
    fig = gcf;

    % Set figure size with ratio
    set(fig, 'Units','pixels');
    set(fig, 'Position',[0 0 width height]);

    hold on;
    for i=1:18
        [color1]=color_plot_cells(i,load_number,cell_coor,velocity);
    end

    for i=1:7
        [color2]=color_plot_line(i,load_number,line_coor,IFTime);
    end
    hold off;

    % Apply window for saved plot
    xlim(zoom_x)
    ylim(zoom_y)

    % SA naming
    for sa = 0:9
        x = geometry(sa+1).distance_from_edge;
        y = geometry(sa+1).height;

        if mod(sa,2) == 0
            x_txt = x - dx;
            y_txt = y - dy;
            hAlign = 'right';
            vAlign = 'top';
```

```
        else
            x_txt = x - dx;
            y_txt = y + dy;
            hAlign = 'right';
            vAlign = 'bottom';
        end

        text(x_txt, y_txt, sprintf('SA%d', sa), 'FontSize', 14, ...
            'FontWeight', 'bold', 'HorizontalAlignment', hAlign, ...
            'VerticalAlignment', vAlign, 'Color', [0 0 1]);
    end

    filename = sprintf('S10H1D_Vel_TOMO_%d.jpg', (load_number - 1) * 50); %Change .jpg or .svg
    saveas(fig, filename);
    close(fig);
end
```

range of color bar

END

Published with MATLAB® R2022b

Appendix III.6.2 – Colour Plot

Appendix III.6.2.1 – color_ plots_cells.m

Note: Only used for the UPV calculations.

```
function [color]=color_plot_cells(cell_n,load_level_number,coor_cell,velocity)
```

Color Calculation p2v2 - V2.0

v1.0 by Hao Cheng(h.cheng-2@tudelft.nl) v3.0 by Vincent Ponson (v.n.ponson@student.tudelft.nl) find the color based on the selected range Specific code for UPV-Tomography Fields(Cells)

CODE

```
vel_c = velocity(cell_n).velocity;
vel   = vel_c(load_level_number);

% Define the cell colour
if vel>1.0  % range of colour bar, can be set as the input (upepr)
    vel=1.0;  % range of colour bar, can be set as the input (upper)
end
if vel<0.8 % range of colour bar, can be set as the input (lower)
    vel=0.8; % range of colour bar, can be set as the input (lower)
end
color=color_find_tomo(0.8,1.0,vel); % range of color bar, can be set as the input (lower,
upper)

% Define the left-bottom and right-top coordinates of the cell
lb_x = coor_cell(cell_n).left_bottom_x;
lb_y = coor_cell(cell_n).left_bottom_y;
rt_x = coor_cell(cell_n).right_top_x;
rt_y = coor_cell(cell_n).right_top_y;
```

```
% Calculate width and height of the square
width = rt_x - lb_x;
height = rt_y - lb_y;

% Draw the square using the specified coordinates
rectangle('Position', [lb_x, lb_y, width, height], ...
          'EdgeColor', [0,0,0], 'FaceColor', ...
          color, 'LineWidth', 1);
```

END

Published with MATLAB® R2022b

Appendix III.6.2.2 – color_plots_line.m

Note: Only used for the UPV calculations.

```
function [color]=color_plot_line(dt_n,load_level_number,line_coor,IFTime)
```

Color Calculation p1v2 - V2.0

v1.0 by Hao Cheng(h.cheng-2@tudelft.nl) v2.0 by Vincent Ponson (v.n.ponson@student.tudelft.nl) Selects color to each line - Tomography

CODE

```
time_c = IFTime(dt_n).time;
dt   = time_c(load_level_number);

% Define the cell colour
if dt>1.0  % range of colour bar, can be set as the input (upepr)
    dt=1.0;  % range of colour bar, can be set as the input (upper)
end

if dt<0.8 % range of colour bar, can be set as the input (lower)
    dt=0.8; % range of colour bar, can be set as the input (lower)
end
color=color_find_tomo(0.8,1.0,dt); % range of color bar, can be set as the input (lower,
upper)

% Define the left-bottom and right-top coordinates of the cell
lb_x = line_coor(dt_n).left_x;
lb_y = line_coor(dt_n).left_y;
rt_x = line_coor(dt_n).right_x;
rt_y = line_coor(dt_n).right_y;

% Calculate width and height of the square
width = rt_x - lb_x;
height = rt_y - lb_y;

% Draw the square using the specified coordinates
rectangle('Position', [lb_x, lb_y, width, height], ...
          'EdgeColor', [0,0,0], 'FaceColor', ...
          color, 'LineWidth', 1);

end
```

END
Published with MATLAB® R2022b

Appendix III.6.3 – Code to determine the Colour

Appendix III.6.3.1 – color_plots_cells.m

Note: Only used for the UPV calculations.


```
function [color]=color_find_tomo(bottom,top,velocity)
```

v1.0 by Hao Cheng, 11/01/2023, email: h.cheng-2@tudelft.nl
v1.1 adjusted by Vincent Ponson, 20/07/2024, email: v.ponson@student.tudelft.nl find the color based on the selected range red to green: typical colors used for numerical simulation

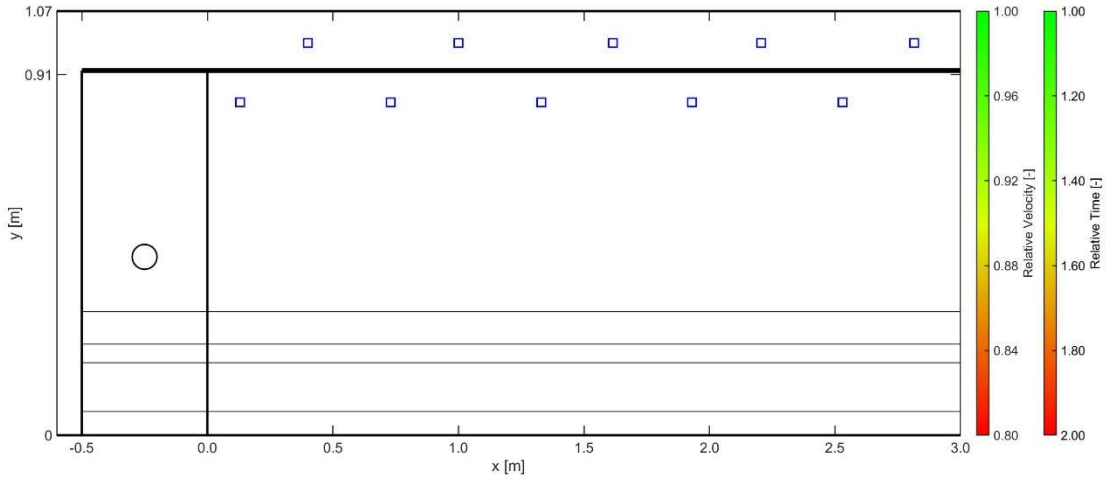
CODE - 2 Colours

```
if velocity < bottom
    velocity = bottom;
elseif velocity > top
    velocity = top;
end

% Divide the range into two segments
delta = (top - bottom) / 2;

if velocity < bottom + delta
    % Transition from red to yellow
    t = (velocity - bottom) / delta;
    color = [1, 0, 0] * (1 - t) + [1, 1, 0] * t; % Red to yellow
else
    % Transition from yellow to green
    t = (velocity - (bottom + delta)) / delta;
    color = [1, 1, 0] * (1 - t) + [0, 1, 0] * t; % Yellow to green
end

end
```

END
Published with MATLAB® R2022b

Appendix III.6.4 – Empty Figures

Appendix III.6.4.1 – empty_figure_.fig



Appendix III.7 – Tomography Plotting Code

Appendix III.7.1 – Figure Plotter

Appendix III.7.1.1 – plot_figure.m

Note: this version is for the whole DIC figure modifier

DIC Figures editor - V1.2

v1.2 by Vincent Ponson (v.n.ponson@student.tudelft.nl)

Adjust and modify DIC figures

plot fields on to DIC

Define window and ratio of the saved figure

```
zoom_x = [0.5 3.5];
zoom_y = [0.75 1.07];

width  = 2000;
height = width * (1/5);

dx = 0.015;
dy = 0.015;

imagenumber = 33; % CHANGE

x_shift = -0.1;
y_shift = -1.06;
x_scale = 0.0067;
y_scale = 0.0065;

for i = 1:imagenumber
    % Read image
    img = imread(sprintf('DIC Stage%d.jpg', i));
```

```
% Calculate physical dimensions
img_width = size(img, 2) * x_scale;
img_height = size(img, 1) * y_scale;

% Create figure
fig = figure;

% Display image with correct scaling and positioning
imshow(img, 'XData', [x_shift, x_shift + img_width], ...
    'YData', [y_shift + img_height, y_shift]);
set(gca, 'YDir', 'normal');  % Cartesian Y-direction (bottom-left origin)

% Set figure size with ratio
set(fig, 'Units','pixels');
set(fig, 'Position',[0 0 width height]);

hold on;

% Apply window for saved plot
xlim(zoom_x);
ylim(zoom_y);

for SA_n = 1:length(geometry)
    color_plot_SA(SA_n, geometry);
end

for RP = 1:24
    color_plot_RP2(RP,geometry);
end

% SA naming
for sa = 0:9
    x = geometry(sa+1).distance_from_edge;
    y = geometry(sa+1).height;

    if mod(sa,2) == 0
        x_txt = x - dx;
        y_txt = y - dy;
        hAlign = 'right';
        vAlign = 'top';
    else
        x_txt = x - dx;
        y_txt = y + dy;
        hAlign = 'right';
        vAlign = 'bottom';
    end

    text(x_txt, y_txt, sprintf('SA%d', sa), 'FontSize', 14, ...
```

```
            'FontWeight', 'bold', 'HorizontalAlignment', hAlign, ...
            'VerticalAlignment', vAlign, 'Color', [0.9 0.9 0.9]);
    end

    % Plot each cell outline
    for cell_n = 1:length(cell_coor)
        color_plot_DIC(cell_n, cell_coor);
    end

    filename = sprintf('S10H1D_DIC_%d.jpg', i);
    exportgraphics(gca, filename, 'Resolution', 300);
    close(gcf);
end
```

END

Appendix III.7.1.2 – plot_DIC.m

Note: this version is for the Group version DIC figure modifier

DIC Figures editor - V1.2

v1.2 by Vincent Ponson (v.n.ponson@student.tudelft.nl)

Adjust and modify DIC figures - Group view

plot fields on to DIC

```
imagenumber = 33; % CHANGE

RP_id = [[1,1,2,3,2,4,3,4];[3,3,4,5,4,6,5,6];[5,5,6,7,6,8,7,8];[7,7,8,9,8,10,9,10]];

x_shift = -0.1;
y_shift = -1.06;
x_scale = 0.0067;
y_scale = 0.0065;

dx = 0.015;   % horizontal text offset
dy = 0.015;   % vertical text offset

% index of Group to zoom in on
sa_idx1 = 7; %first SA of the Group, [1 3 5 7]
sa_idx2 = 10; %last SA of the Group, [4 6 8 10]

x1 = geometry(sa_idx1).distance_from_edge;
y1 = geometry(sa_idx1).height;

x2 = geometry(sa_idx2).distance_from_edge;
y2 = geometry(sa_idx2).height;

w = 0.15;

zoom_x = [x1 - w, x2 + w];
zoom_y = [y1 - w*3/4, y2 + w];
```

```
for i = 1:imagenumber
    % Read image
    img = imread(sprintf('DIC Stage%d.jpg', i));

    % Calculate physical dimensions
    img_width  = size(img, 2) * x_scale;
    img_height = size(img, 1) * y_scale;

    % Create figure
    fig_zoom = figure;
    set(fig_zoom, 'PaperUnits', 'inches', 'PaperPosition', [0 0 5 1.75]);

    % Display image again
    imshow(img, ...
        'XData', [x_shift, x_shift + img_width], ...
        'YData', [y_shift + img_height, y_shift]);
    set(gca, 'YDir', 'normal');
    hold on;

    % Plot same cells, SA points & Ray Paths
    for cell_n = 1:length(cell_coor)
        color_plot_DIC(cell_n, cell_coor);
    end

    for SA_n = 1:length(geometry)
        color_plot_SA(SA_n, geometry);
    end

    color_plot_RP(sa_idx1,RP_id,geometry);

    % SA naming
    for sa = 0:9
        x = geometry(sa+1).distance_from_edge;
        y = geometry(sa+1).height;

        if mod(sa,2) == 0
            x_txt = x - dx;
            y_txt = y - dy;
            hAlign = 'right';
            vAlign = 'top';
        else
            x_txt = x - dx;
            y_txt = y + dy;
            hAlign = 'right';
            vAlign = 'bottom';
        end

        text(x_txt, y_txt, sprintf('SA%d', sa), 'FontSize', 10, ...
            'FontWeight', 'bold', 'HorizontalAlignment', hAlign, ...
```

```
            'VerticalAlignment', vAlign, 'Color', [0.9 0.9 0.9]);
    end

    % Set axis limits to zoom area
    xlim(zoom_x);
    ylim(zoom_y);

    % Save the zoomed figure
    filename_zoom = sprintf('S10H1D_DIC_Group4_%d.jpg', i);
    saveas(fig_zoom, filename_zoom);
    close(fig_zoom);
end
```

END

Appendix III.7.1.3 – plot_DIC_TOMO.m

Note: this version is for the tomography Group version DIC figure modifier

DIC Figures editter - V1.2

v1.2 by Vincent Ponson (v.n.ponson@student.tudelft.nl)

Adjust and modify DIC figures - Tomography Group view

plot fields on to DIC

```
imagenumber = 33; % CHANGE

x_shift = -0.1;
y_shift = -1.06;
x_scale = 0.0067;
y_scale = 0.0065;

dx = 0.015;   % horizontal text offset
dy = 0.015;   % vertical text offset

% index of Group to zoom in on
sa_idx1 = 6; %first SA of the Group, [2 4 6]
sa_idx2 = 9; %last SA of the Group, [5 7 9]

x1 = geometry(sa_idx1).distance_from_edge;
y1 = geometry(sa_idx1).height;

x2 = geometry(sa_idx2).distance_from_edge;
y2 = geometry(sa_idx2).height;

w = 0.15;

zoom_x = [x1 - w, x2 + w];
zoom_y = [y2 - w*3/4, y1 + w];

for i = 1:imagenumber
    % Read image
    img = imread(sprintf('DIC Stage%d.jpg', i));
```

```
% Calculate physical dimensions
img_width  = size(img, 2) * x_scale;
img_height = size(img, 1) * y_scale;

% Create figure
fig_zoom = figure;
set(fig_zoom, 'PaperUnits', 'inches', 'PaperPosition', [0 0 5 1.75]);

% Display image again
imshow(img, ...
    'XData', [x_shift, x_shift + img_width], ...
    'YData', [y_shift + img_height, y_shift]);
set(gca, 'YDir', 'normal');
hold on;

% Plot same cells & SA points
for cell_n = 1:length(cell_coor)
    color_plot_DIC(cell_n, cell_coor);
end

for SA_n = 1:length(geometry)
    color_plot_SA(SA_n, geometry);
end

% SA naming
for sa = 0:9
    x = geometry(sa+1).distance_from_edge;
    y = geometry(sa+1).height;

    if mod(sa,2) == 0
        x_txt = x - dx;
        y_txt = y - dy;
        hAlign = 'right';
        vAlign = 'top';
    else
        x_txt = x - dx;
        y_txt = y + dy;
        hAlign = 'right';
        vAlign = 'bottom';
    end

    text(x_txt, y_txt, sprintf('SA%d', sa), 'FontSize', 10, ...
        'FontWeight', 'bold', 'HorizontalAlignment', hAlign, ...
        'VerticalAlignment', vAlign, 'Color', [0.9 0.9 0.9]);
end

% Set axis limits to zoom area
xlim(zoom_x);
```

```
    ylim(zoom_y);

    % Save the zoomed figure
    filename_zoom = sprintf('S10H1D_DIC_Tomo_Group3_%d.jpg', i);
    saveas(fig_zoom, filename_zoom);
    close(fig_zoom);
end
```

END

Appendix III.7.2 – Object Plotter

Appendix III.7.2.1 – color_plot_SA.m

Note: this version is for both DIC figure modifier

SA plotter - V1.0

v1.0 by Vincent Ponson (v.n.ponson@student.tudelft.nl)
Helps plot the Smart Aggregates to the image

```
function [color]=color_plot_SA(SA_n,geometry)
```

CODE

Define border color Cell boundaries

```
    x = geometry(SA_n).distance_from_edge;
    y = geometry(SA_n).height;

    w = 0.01;

    lb_x = x-w/2;
    lb_y = y - w/2;

    % Draw border-only rectangle
    rectangle('Position', [lb_x, lb_y, w, w], ...
              'EdgeColor', [0.9 0.9 0.9], 'LineWidth', 0.95, ...
              'FaceColor', [0.9 0.9 0.9]);  % Ensure no fill

end
```

END

Appendix III.7.2.2 – color_plot_DIC.m

Note: this version is for both DIC figure modifier

DIC Figures editor - V1.2

v1.2 by Vincent Ponson (v.n.ponson@student.tudelft.nl)
Adjust and modify DIC figures - Group view

```
function [color]=color_plot_DIC(cell_n,coor_cell)
```

CODE

Define border color Cell boundaries

```
    lb_x = coor_cell(cell_n).left_bottom_x;
```

```
    lb_y = coor_cell(cell_n).left_bottom_y;
    rt_x = coor_cell(cell_n).right_top_x;
    rt_y = coor_cell(cell_n).right_top_y;

    % Calculate width and height
    width = rt_x - lb_x;
    height = rt_y - lb_y;

    % Draw border-only rectangle (no fill)
    rectangle('Position', [lb_x, lb_y, width, height], ...
              'EdgeColor', [0.9, 0.9, 0.9], 'LineWidth', 0.5, ...
              'FaceColor', 'none');  % Ensure no fill
end
```

END

Appendix III.7.2.3 – color_plot_RP.m

Note: this version is for the Group and tomography Group version DIC figure modifier

Ray-path Plotter - V2.0

v2.0 by Vincent Ponson (v.n.ponson@student.tudelft.nl)
Helps plot the SA ray-paths to the image Group version

```
function [color]=color_plot_RP(sa_idx1,RP_id,geometry)
```

CODE

```
n = (sa_idx1 + 1) / 2;
N = RP_id(n, :);

x = zeros(1,8);
y = zeros(1,8);

for i = 1:8
    N_id = N(i);
    x(i) = geometry(N_id).distance_from_edge;
    y(i) = geometry(N_id).height;
end

% Draw ray paths (pairwise)
line([x(1) x(5)], [y(1) y(5)], 'Color', [0.9 0.9 0.9], 'LineWidth', 0.25);
line([x(2) x(6)], [y(2) y(6)], 'Color', [0.9 0.9 0.9], 'LineWidth', 0.25);
line([x(3) x(7)], [y(3) y(7)], 'Color', [0.9 0.9 0.9], 'LineWidth', 0.25);
line([x(4) x(8)], [y(4) y(8)], 'Color', [0.9 0.9 0.9], 'LineWidth', 0.25);

color = [];
end
```

END

Appendix III.7.2.4 – color_plot_RP2.m

Note: this version is for the whole version DIC figure modifier

Ray-path Plotter - V2.0

v2.0 by Vincent Ponson Helps plot the SA ray-paths to the image Full DIC version

```
function [color]=color_plot_RP2(i,geometry)
```

CODE

```
list_1 = [0,0,0, 1,1,1, 2,2,2, 3,3,3, 4,4,4, 5,5,5, 6,6,6, 7,7, 8];
list_2 = [1,2,3, 2,3,4, 3,4,5, 4,5,6, 5,6,7, 6,7,8, 7,8,9, 8,9, 9];

l1 = list_1(i)+1;
l2 = list_2(i)+1;

x1 = geometry(l1).distance_from_edge;
y1 = geometry(l1).height;

x2 = geometry(l2).distance_from_edge;
y2 = geometry(l2).height;

% Draw ray paths (pairwise)
line([x1 x2], [y1 y2], 'Color', [0.9 0.9 0.9], 'LineWidth', 0.5);
color = [];
end
```

END

Appendix IV – Group Analysis

Appendix III is organized as follows:

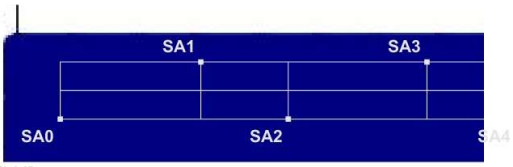
- **Appendix IV.1** – Group Analysis of beam S10H2D
- **Appendix IV.2** – Group Analysis of beam S10H2A
- **Appendix IV.3** – Group Analysis of beam S10H1D

Legend for the Group Analysis Tables:

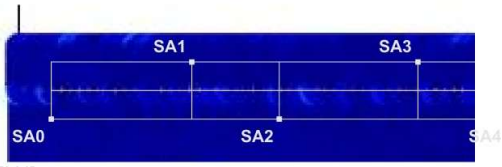
	Meaning	abbreviation	Meaning
Num.	DIC Load Steps	F-	Field
S	Sign of cracking	I - XVIII	Field number
Cs	Clear Sign of cracking	dt-	Interface
Fc	Flexural crack	1 - 7	Interface number
Sc	Shear crack	F-	Field
D	Delamination		

Appendix IV.1 – Results of beam S10H2D

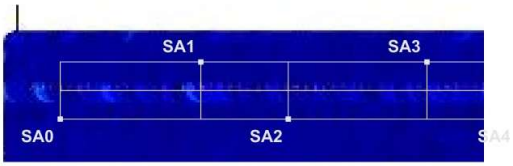
Group 1 – Group DIC Figures and Overall Analysis



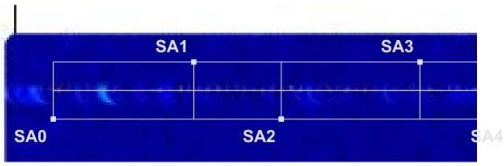
(a)-0 [kN]



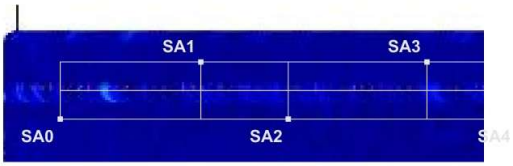
(b)-50 [kN]



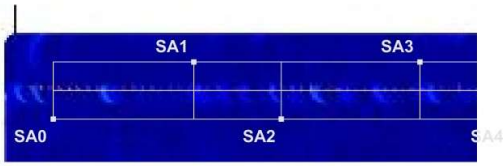
(c)-100 [kN]



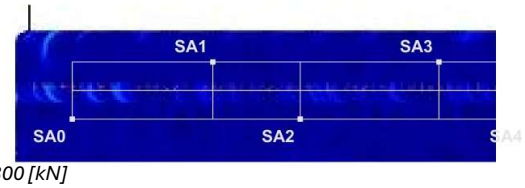
(d)-150 [kN]



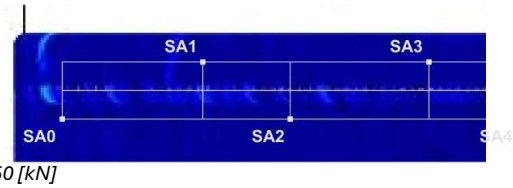
(e)-200 [kN]



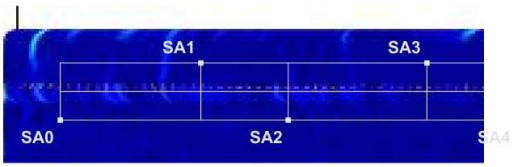
(f)-250 [kN]



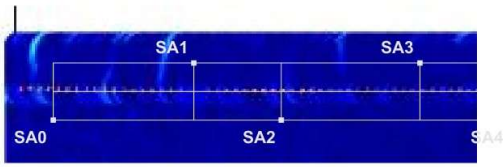
(g)-300 [kN]



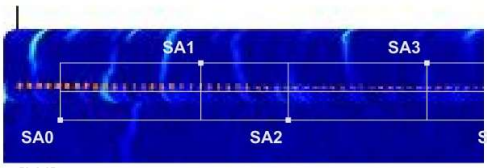
(h)-350 [kN]



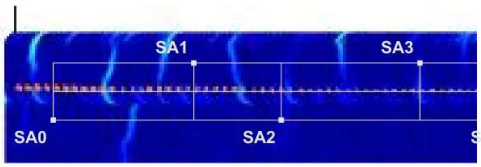
(i)-400 [kN]



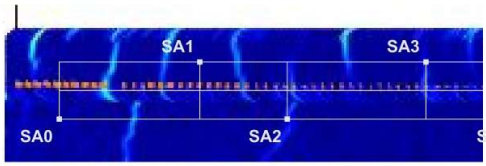
(j)-450 [kN]



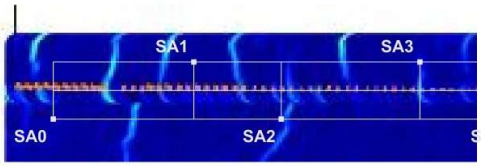
(k)-550 [kN]



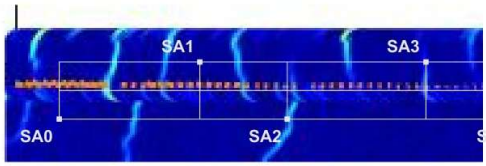
(l)-600 [kN]



(m)-650 [kN]



(n)-700 [kN]

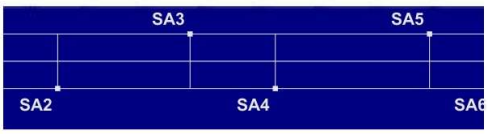


(o)-750 [kN]

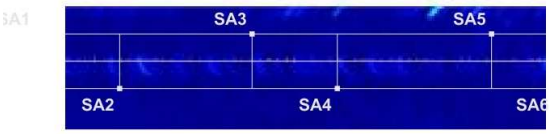
Group 1 – Overall Analysis (p1v2)						
Num. [-]	Load DIC [kN]	Load [kN]	Flexural Cracks		Shear Cracks	
1	0,2	0				
2	47,3	50				
3	100,2	100				
4	149,1	150				
5	196,9	200				
6	249,2	250				
7	300,2	300				
8	348,7	350	S, Fc F-I			
9	400,4	400	S, Fc F-I			
10	448,1	450	S, Fc F-I(2)&V	& Cs Fc F-I		
-	-	500	-	-	-	-
11	552,4	550	CS, Fc F-I&V	& S Fc F-III	S, Sc F-II	
12	600,4	600			CS, Sc F-II	
13	650,8	650	S, Fc F-V(2)		S, Sc F-VI	
14	700,7	700	CS, Fc F-III		SC, Sc F-VI	
15	751,2	750				

Group 1 – Overall Analysis (p2v2)						
Num. [-]	Load DIC [kN]	Load [kN]	Delamination		Other Cracks	
1	0,2	0				
2	47,3	50				
3	100,2	100				
4	149,1	150				
5	196,9	200				
6	249,2	250				
7	300,2	300			S, Fc out L-F-I	
8	348,7	350				
9	400,4	400				
10	448,1	450	S, D dt1&2	& L-dt1		
-	-	500	-	-	-	-
11	552,4	550	CS, dt1	& L-dt1		
12	600,4	600				
13	650,8	650	CS, dt2			
14	700,7	700			S, Sc L-F-I	
15	751,2	750			CS, Sc L-F-I	

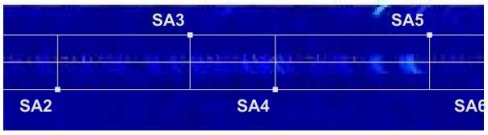
Group 2 – Group DIC Figures and Overall Analysis



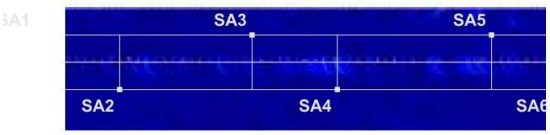
(a)-0 [kN]



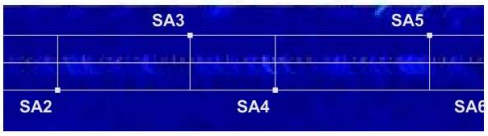
(b)-50 [kN]



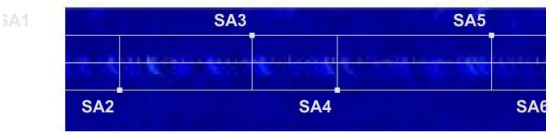
(c)-100 [kN]



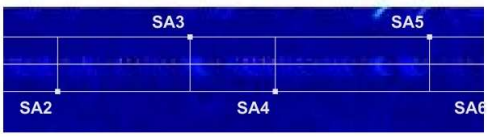
(d)-150 [kN]

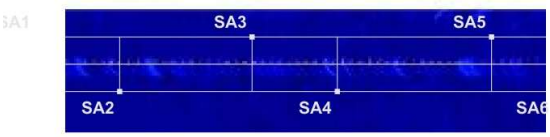


(e)-200 [kN]

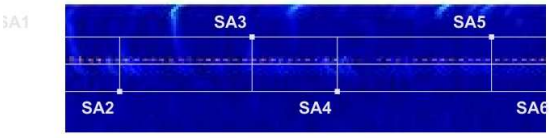


(f)-250 [kN]

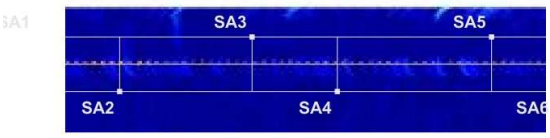




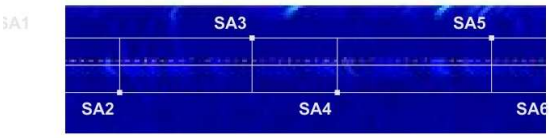
(g)-300 [kN]



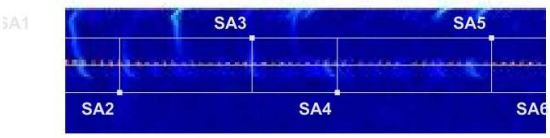
(h)-350 [kN]



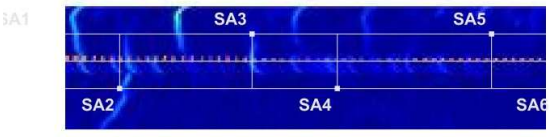
(i)-400 [kN]



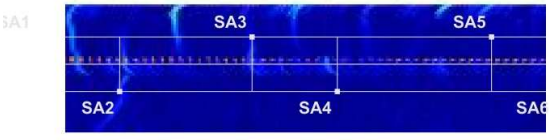
(j)-450 [kN]



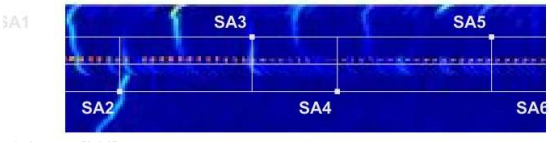
(k)-550 [kN]



(l)-600 [kN]

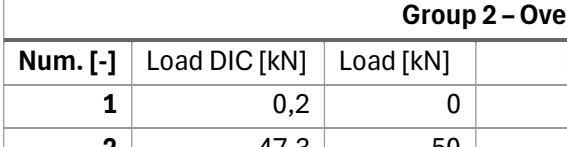


(m)-650 [kN]



(n)-700 [kN]

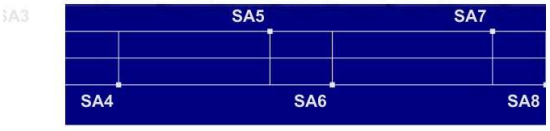
(o)-750 [kN]



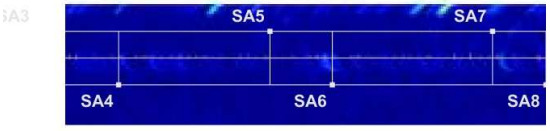
Group 2 – Overall Analysis (p1v2)						
Num. [-]	Load DIC [kN]	Load [kN]	Flexural Cracks		Shear Cracks	
1	0,2	0				
2	47,3	50				
3	100,2	100				
4	149,1	150				
5	196,9	200				
6	249,2	250				
7	300,2	300				
8	348,7	350				
9	400,4	400				
10	448,1	450				
-	-	500	-	-	-	-
11	552,4	550				
12	600,4	600	S, Fc F-VII			
13	650,8	650	S, Fc F-IX		S, Sc F-VI	
14	700,7	700	CS, Fc F-VII	CS, Fc F-IX	CS, Sc F-VI	
15	751,2	750				

Group 2 – Overall Analysis (p2v2)						
Num. [-]	Load DIC [kN]	Load [kN]	Delamination		Other Cracks	
1	0,2	0				
2	47,3	50				
3	100,2	100				
4	149,1	150				
5	196,9	200				
6	249,2	250				
7	300,2	300				
8	348,7	350				
9	400,4	400				
10	448,1	450	S, D dt2		S, Fc F-V	
-	-	500	-	-	-	-
11	552,4	550	S, D dt2&3&4		CS, Fc F-V	
12	600,4	600			CS, Fc F-III	
13	650,8	650	CS, D dt2			
14	700,7	700				
15	751,2	750	CS, dt5			

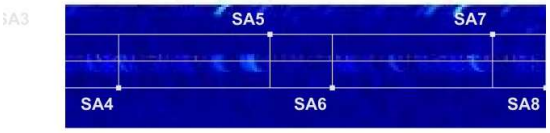
Group 3 – Group DIC Figures and Overall Analysis



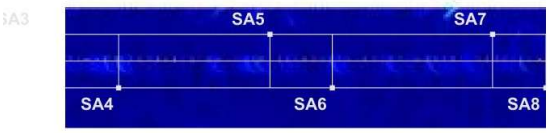
(a)-0 [kN]



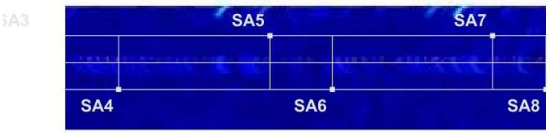
(b)-50 [kN]



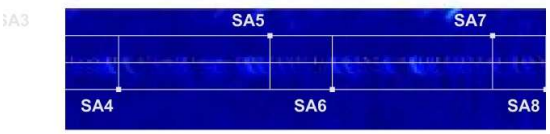
(c)-100 [kN]



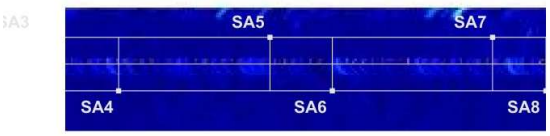
(d)-150 [kN]

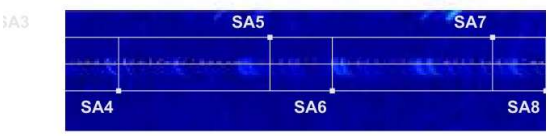


(e)-200 [kN]

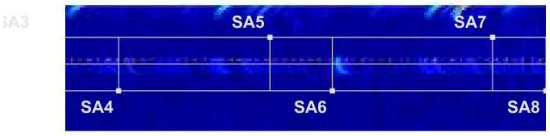


(f)-250 [kN]

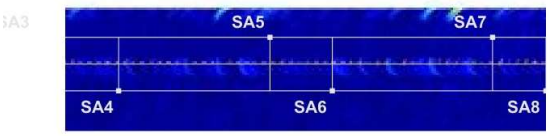




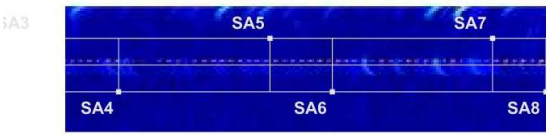
(g)-300 [kN]



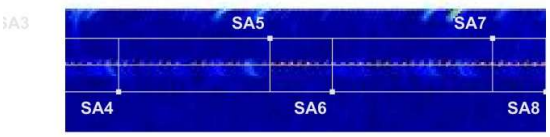
(h)-350 [kN]



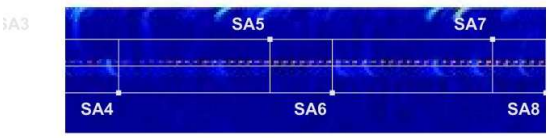
(i)-400 [kN]



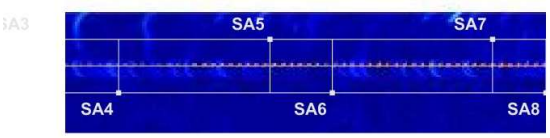
(j)-450 [kN]



(k)-550 [kN]



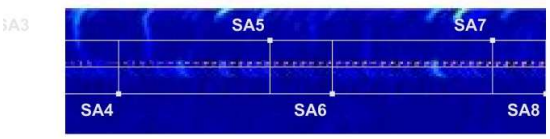
(l)-600 [kN]



(m)-650 [kN]

(n)-700 [kN]

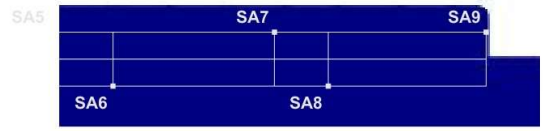
(o)-750 [kN]



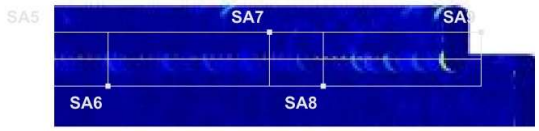
Group 3 – Overall Analysis (p1v2)						
Num. [-]	Load DIC [kN]	Load [kN]	Flexural Cracks		Shear Cracks	
1	0,2	0				
2	47,3	50				
3	100,2	100				
4	149,1	150				
5	196,9	200				
6	249,2	250				
7	300,2	300				
8	348,7	350				
9	400,4	400				
10	448,1	450				
-	-	500	-	-	-	-
11	552,4	550				
12	600,4	600				
13	650,8	650				
14	700,7	700				
15	751,2	750				

Group 3 – Overall Analysis (p2v2)						
Num. [-]	Load DIC [kN]	Load [kN]	Delamination		Other Cracks	
1	0,2	0				
2	47,3	50				
3	100,2	100				
4	149,1	150				
5	196,9	200				
6	249,2	250				
7	300,2	300				
8	348,7	350				
9	400,4	400				
10	448,1	450				
-	-	500	-	-	-	-
11	552,4	550	S, D dt5,6&7		S, D dt4&R-dt7	
12	600,4	600				
13	650,8	650	CS, D dt5,6&7		CS, D dt4&R-dt7	S, Fc F-IX
14	700,7	700				
15	751,2	750				

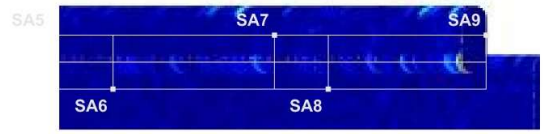
Group 4 – Group DIC Figures and Overall Analysis



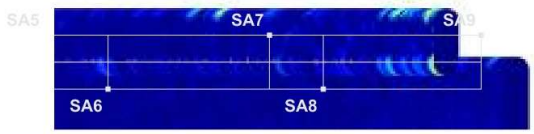
(a)-0 [kN]



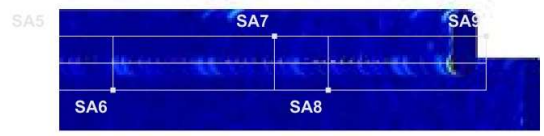
(b)-50 [kN]



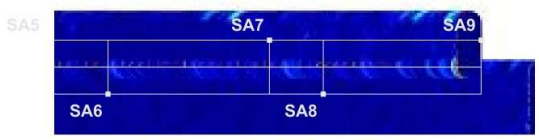
(c)-100 [kN]



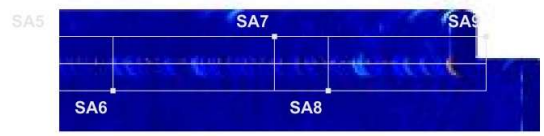
(d)-150 [kN]

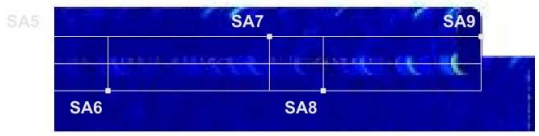


(e)-200 [kN]

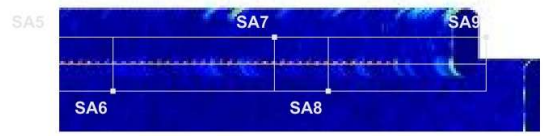


(f)-250 [kN]

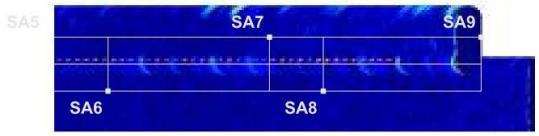




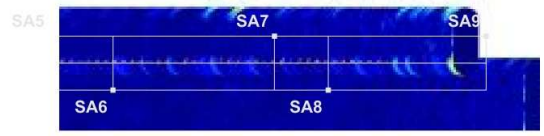
(g)-300 [kN]



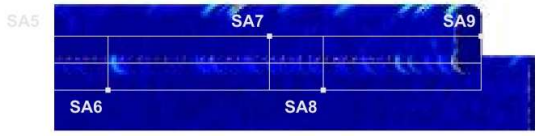
(h)-350 [kN]



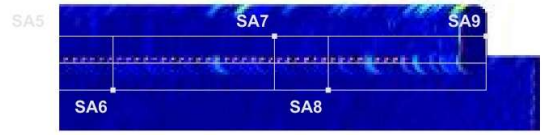
(i)-400 [kN]



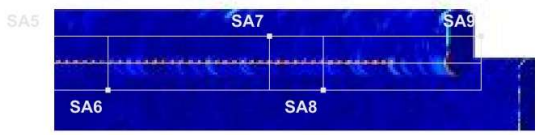
(j)-450 [kN]



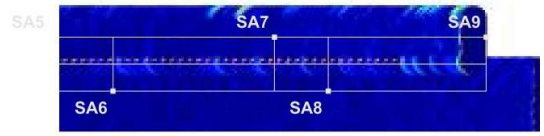
(k)-550 [kN]



(l)-600 [kN]

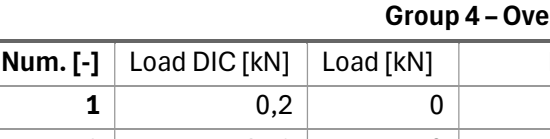


(m)-650 [kN]



(n)-700 [kN]

(o)-750 [kN]

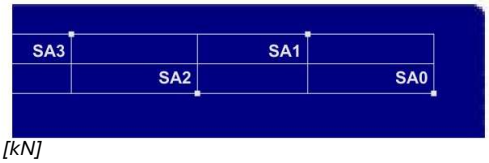


Group 4 – Overall Analysis (p1v2)						
Num. [-]	Load DIC [kN]	Load [kN]	Flexural Cracks		Shear Cracks	
1	0,2	0				
2	47,3	50				
3	100,2	100				
4	149,1	150				
5	196,9	200				
6	249,2	250				
7	300,2	300				
8	348,7	350				
9	400,4	400				
10	448,1	450	S, Fc F-XVII			
-	-	500	-	-	-	-
11	552,4	550				
12	600,4	600				
13	650,8	650				
14	700,7	700	CS, Fc F-XVIII			
15	751,2	750				

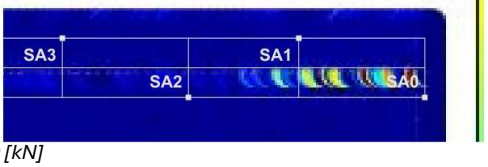
Group 4 – Overall Analysis (p2v2)						
Num. [-]	Load DIC [kN]	Load [kN]	Delamination		Other Cracks	
1	0,2	0				
2	47,3	50				
3	100,2	100				
4	149,1	150				
5	196,9	200				
6	249,2	250				
7	300,2	300				
8	348,7	350				
9	400,4	400				
10	448,1	450				
-	-	500	-	-	-	-
11	552,4	550	S, D dt6&7		S, D R-dt7	
12	600,4	600				
13	650,8	650				
14	700,7	700				
15	751,2	750				

Appendix IV.2 – Results of beam S10H2A

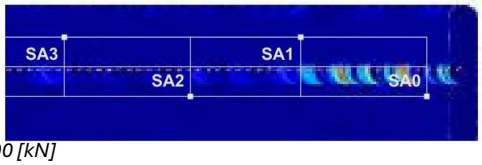
Group 1 – Group DIC Figures and Overall Analysis



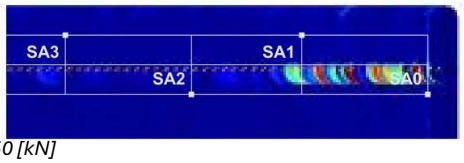
(a)-0 [kN]



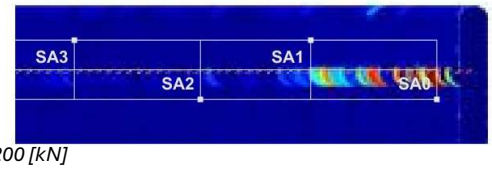
(b)-50 [kN]



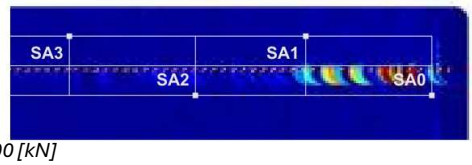
(c)-100 [kN]



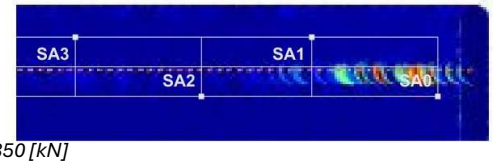
(d)-150 [kN]



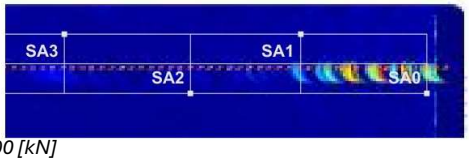
(e)-200 [kN]



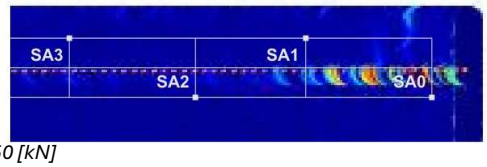
(f)-300 [kN]



(g)-350 [kN]

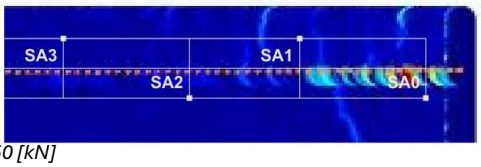


(h)-400 [kN]

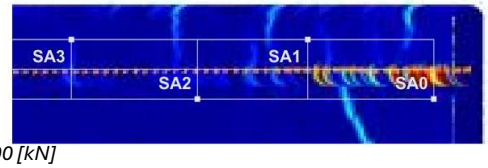


(i)-550 [kN]

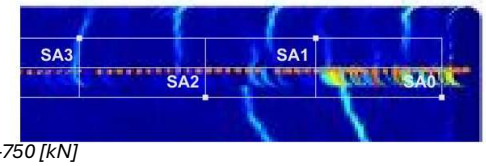
(j)-600 [kN]



(k)-650 [kN]



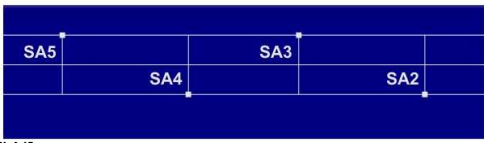
(l)-700 [kN]



(m)-750 [kN]

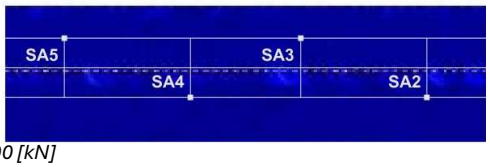
Group 1 – Overall Analysis (p1v2)						
Num. [-]	Load DIC [kN]	Load [kN]	Flexural Cracks		Shear Cracks	
1	0,1	0				
2	49,5	50				
3	99,0	100				
4	151,5	150				
5	201,2	200				
-	-	250	-	-	-	-
6	300,2	300				
7	353,6	350				
8	401,4	400				
-	-	450	-	-	-	-
-	-	500	-	-	-	-
9	551,1	550	S, Fc F-I			
10	609,8	600				
11	650,4	650	Cs, FC F-I&II	S, Fc F-I (2)	Cs, Sc F-II	
15	690,4	700	Cs, Fc F-V			
20	751,8	750	Cs, Fc F-V(2)		Cs, Sc F-IV	

Group 1 – Overall Analysis (p2v2)						
Num. [-]	Load DIC [kN]	Load [kN]	Delamination		Other Cracks	
1	0,1	0				
2	49,5	50				
3	99,0	100				
4	151,5	150				
5	201,2	200				
-	-	250	-	-	-	-
6	300,2	300	S, D dt 1&2&L-dt1			
7	353,6	350				
8	401,4	400				
-	-	450	-	-	-	-
-	-	500	-	-	-	-
9	551,1	550	CS, D dt 1&2&L-dt1			
10	609,8	600				
11	650,4	650				
15	690,4	700				
20	751,8	750				

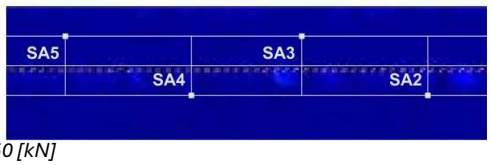


(a)-0 [kN]

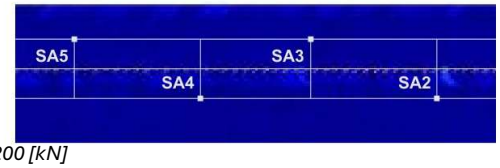
(b)-50 [kN]



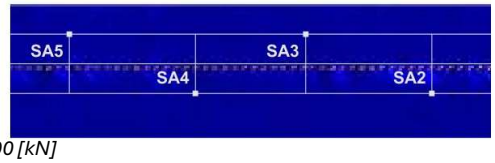
(c)-100 [kN]



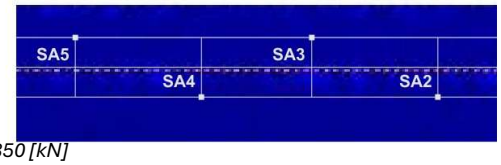
(d)-150 [kN]



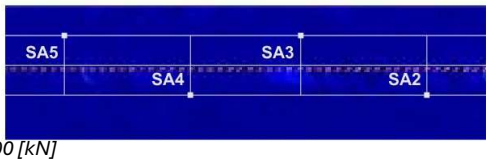
(e)-200 [kN]



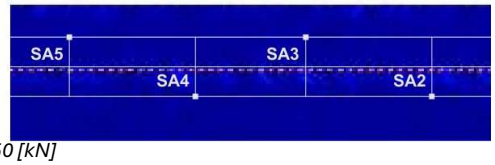
(f)-300 [kN]



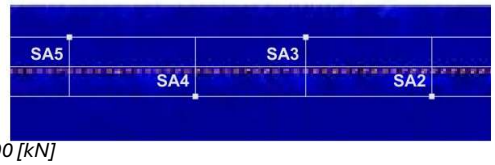
(g)-350 [kN]



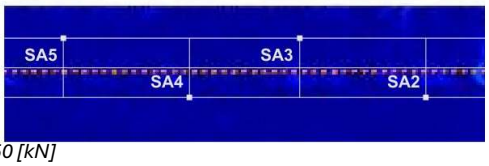
(h)-400 [kN]



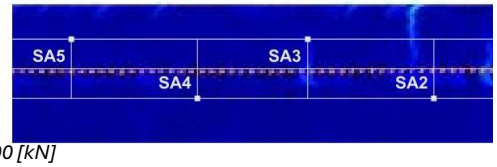
(i)-550 [kN]



(j)-600 [kN]



(k)-650 [kN]



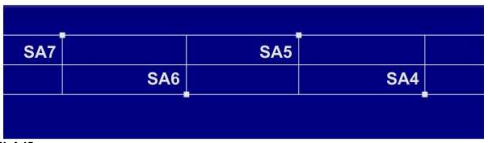
(l)-700 [kN]

(m)-750 [kN]

Group 2 – Overall Analysis (p1v2)						
Num. [-]	Load DIC [kN]	Load [kN]	Flexural Cracks		Shear Cracks	
1	0,1	0				
2	49,5	50				
3	99,0	100				
4	151,5	150				
5	201,2	200				
-	-	250	-	-	-	-
6	300,2	300				
7	353,6	350				
8	401,4	400				
-	-	450	-	-	-	-
-	-	500	-	-	-	-
9	551,1	550				
10	609,8	600				
11	650,4	650				
15	690,4	700	Cs, Fc F-V			
20	751,8	750	Cs, Fc F-V(2)		Cs, Sc F-IV	

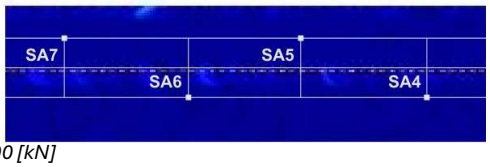
Group 2 – Overall Analysis (p2v2)						
Num. [-]	Load DIC [kN]	Load [kN]	Delamination		Other Cracks	
1	0,1	0				
2	49,5	50				
3	99,0	100				
4	151,5	150				
5	201,2	200				
-	-	250	-	-	-	-
6	300,2	300	S, D dt2,3&4			
7	353,6	350				
8	401,4	400				
-	-	450	-	-	-	-
-	-	500	-	-	-	-
9	551,1	550	Cs, D dt2,3&4			
10	609,8	600				
11	650,4	650				
15	690,4	700				
20	751,8	750				

Group 3 – Group DIC Figures and Overall Analysis

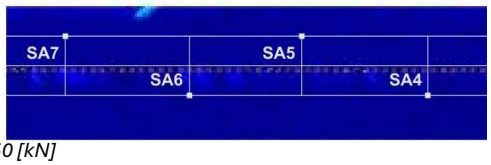


(a)-0 [kN]

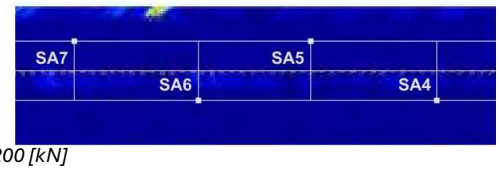
(d)-50 [kN]



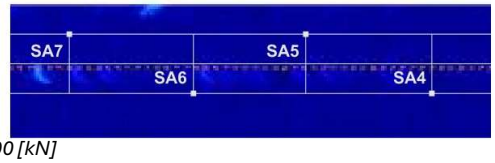
(c)-100 [kN]



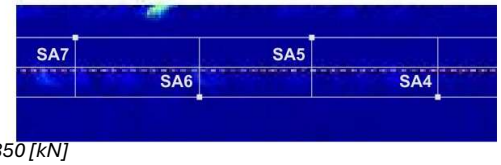
(b)-150 [kN]



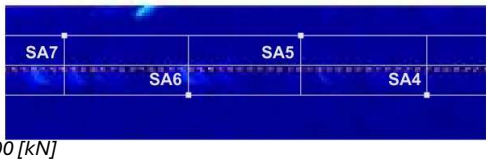
(e)-200 [kN]



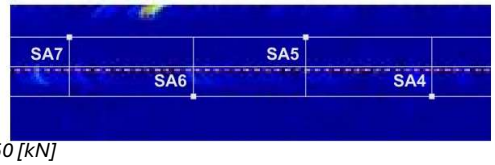
(f)-300 [kN]



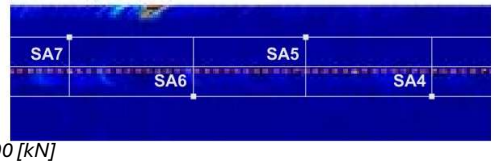
(g)-350 [kN]



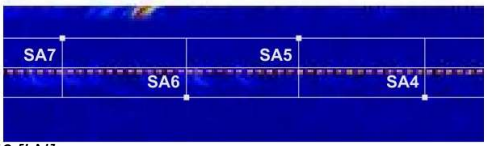
(h)-400 [kN]



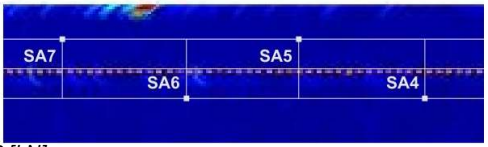
(i)-550 [kN]



(j)-600 [kN]



(k)-650 [kN]

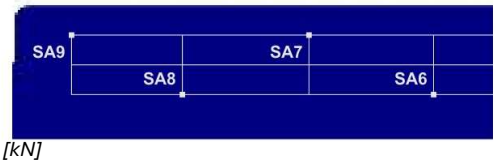


(l)-700 [kN]

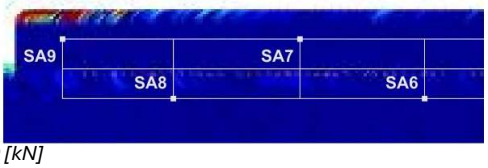
(m)-750 [kN]

Group 3 – Overall Analysis (p1v2)						
Num. [-]	Load DIC [kN]	Load [kN]	Flexural Cracks		Shear Cracks	
1	0,1	0				
2	49,5	50				
3	99,0	100				
4	151,5	150				
5	201,2	200				
-	-	250	-	-	-	-
6	300,2	300				
7	353,6	350				
8	401,4	400				
-	-	450	-	-	-	-
-	-	500	-	-	-	-
9	551,1	550				
10	609,8	600				
11	650,4	650				
15	690,4	700				
20	751,8	750				

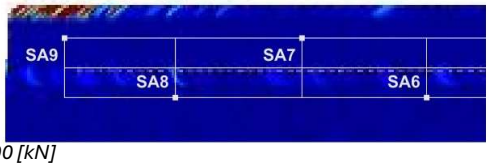
Group 3 – Overall Analysis (p2v2)						
Num. [-]	Load DIC [kN]	Load [kN]	Delamination		Other Cracks	
1	0,1	0				
2	49,5	50				
3	99,0	100				
4	151,5	150				
5	201,2	200				
-	-	250	-	-	-	-
6	300,2	300	S, D dt6,7&R-dt7			
7	353,6	350				
8	401,4	400				
-	-	450	-	-	-	-
-	-	500	-	-	-	-
9	551,1	550	Cs, D dt6,7&R-dt7			
10	609,8	600				
11	650,4	650				
15	690,4	700				
20	751,8	750				



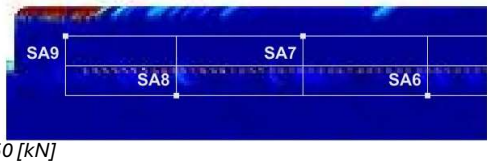
(a)-0 [kN]



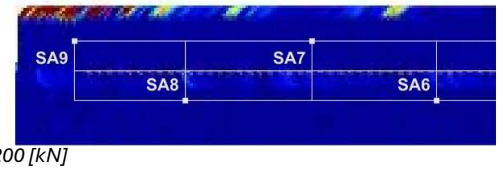
(b)-50 [kN]



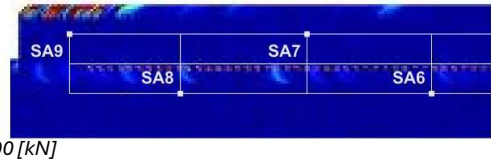
(c)-100 [kN]



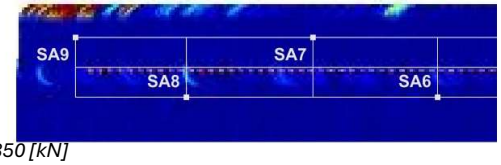
(d)-150 [kN]



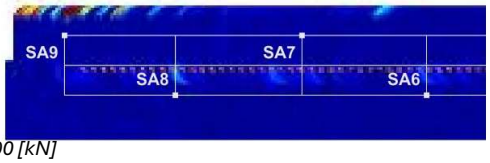
(e)-200 [kN]



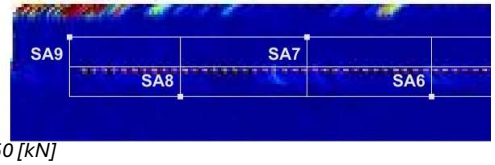
(f)-300 [kN]



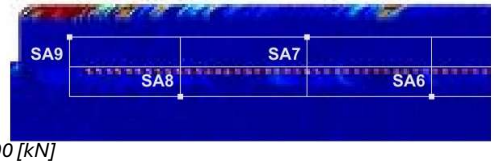
(g)-350 [kN]



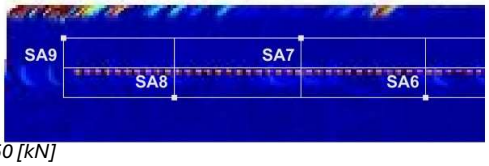
(h)-400 [kN]



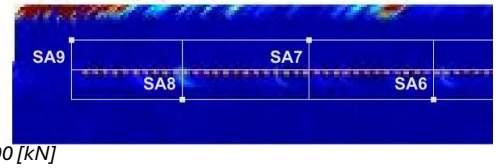
(i)-550 [kN]



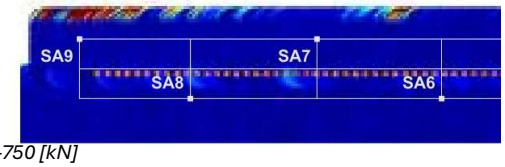
(j)-600 [kN]



(k)-650 [kN]



(l)-700 [kN]



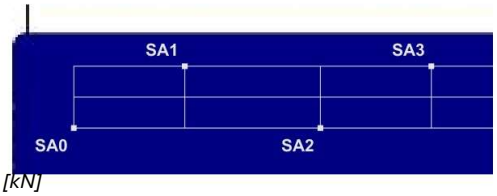
(m)-750 [kN]

Group 4 – Overall Analysis (p1v2)						
Num. [-]	Load DIC [kN]	Load [kN]	Flexural Cracks		Shear Cracks	
1	0,1	0				
2	49,5	50				
3	99,0	100				
4	151,5	150				
5	201,2	200				
-	-	250				
6	300,2	300				
7	353,6	350				
8	401,4	400				
-	-	450				
-	-	500	-	-	-	-
9	551,1	550				
10	609,8	600				
11	650,4	650				
15	690,4	700				
20	751,8	750				

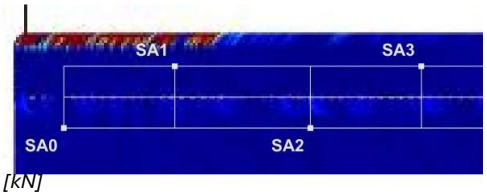
Group 4 – Overall Analysis (p2v2)						
Num. [-]	Load DIC [kN]	Load [kN]	Delamination		Other Cracks	
1	0,1	0				
2	49,5	50				
3	99,0	100				
4	151,5	150				
5	201,2	200				
-	-	250				
6	300,2	300				
7	353,6	350				
8	401,4	400				
-	-	450				
-	-	500				
9	551,1	550				
10	609,8	600				
11	650,4	650				
15	690,4	700				
20	751,8	750				

Appendix IV.3 – Results of beam S10H1D

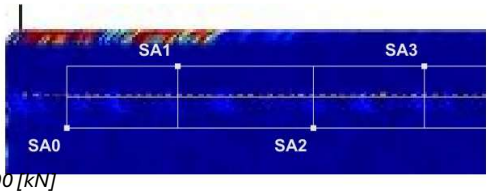
Group 1 – Group DIC Figures and Overall Analysis



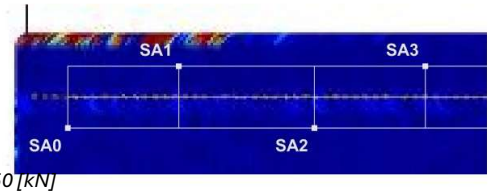
(a)-0 [kN]



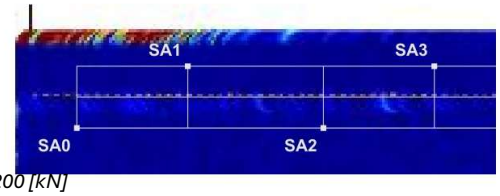
(b)-50 [kN]



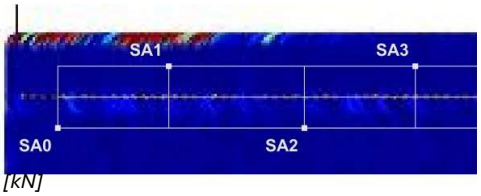
(c)-100 [kN]



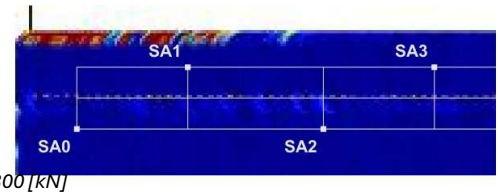
(d)-150 [kN]



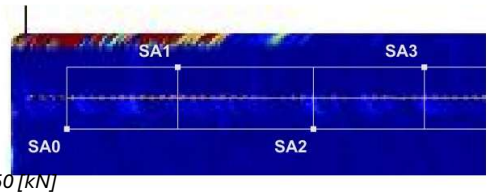
(e)-200 [kN]



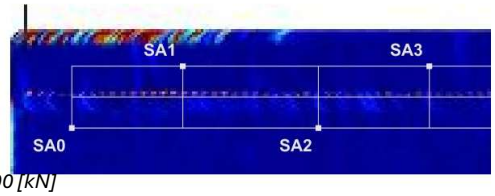
(f)-250 [kN]



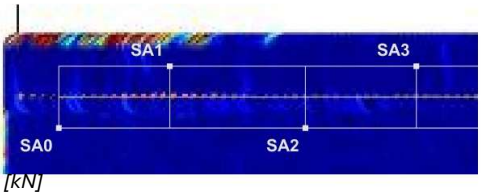
(g)-300 [kN]



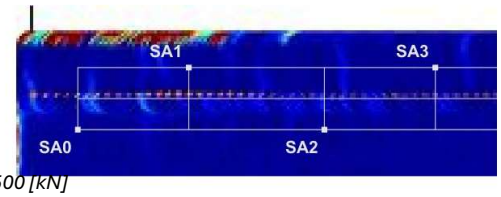
(h)-350 [kN]



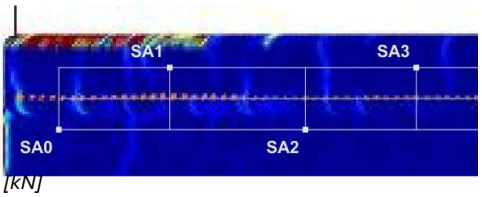
(i)-400 [kN]



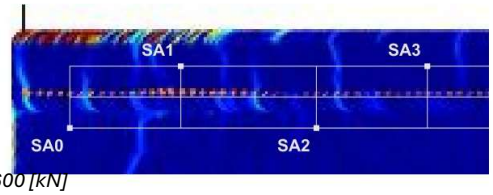
(j)-450 [kN]



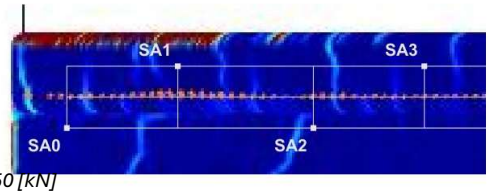
(k)-500 [kN]



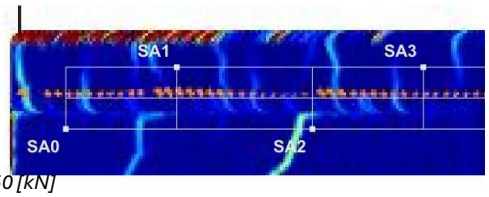
(l)-550 [kN]



(m)-600 [kN]



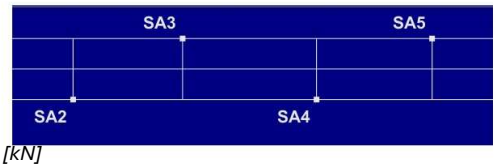
(n)-650 [kN]



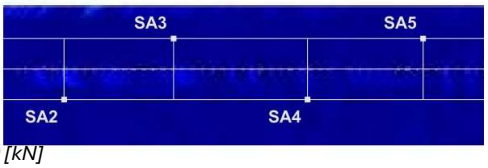
(o)-750 [kN]

Group 1 – Overall Analysis (p1v2)						
Num. [-]	Load DIC [kN]	Load [kN]	Flexural Cracks		Shear Cracks	
1	0,1	0				
2	47,5	50				
3	100,7	100				
4	150,2	150				
5	198,0	200				
6	248,4	250				
7	299,8	300				
8	351,0	350				
9	401,2	400	S, Fc F-I			
10	448,1	450				
11	500,9	500	S, Fc F-III&V	S, Fc F-I (2&3)		
12	549,7	550	Cs, Fc F-I (1&2) &V	S, Fc F-III (2)	S, Sc F-II	
13	598,2	600	Cs, Fc F-III		Cs, Sc F-II	
14	647,9	650	S, Fc F-V (2)		Cs, Sc F-IV	
-	-	700				
15	750,9	750				

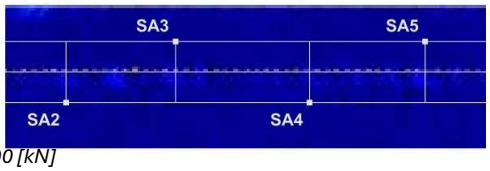
Group 1 – Overall Analysis (p2v2)						
Num. [-]	Load DIC [kN]	Load [kN]	Delamination		Other Cracks	
1	0,1	0				
2	47,5	50				
3	100,7	100				
4	150,2	150				
5	198,0	200				
6	248,4	250				
7	299,8	300				
8	351,0	350				
9	401,2	400	S, D dt1 &L-dt1			
10	448,1	450				
11	500,9	500	Cs, D dt1 &L-dt1	S, D dt2	S, Fc L-F-I	
12	549,7	550				
13	598,2	600				
14	647,9	650				
-	-	700				
15	750,9	750				



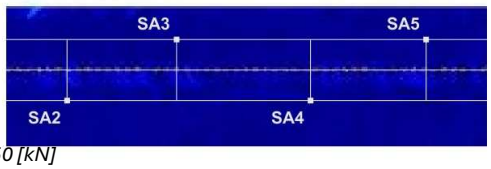
(a)-0 [kN]



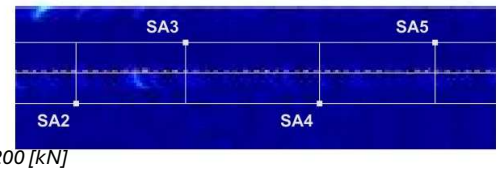
(b)-50 [kN]



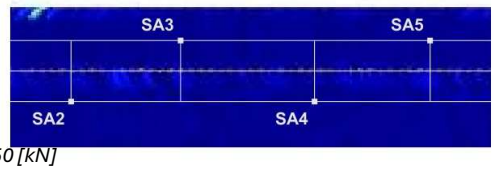
(c)-100 [kN]



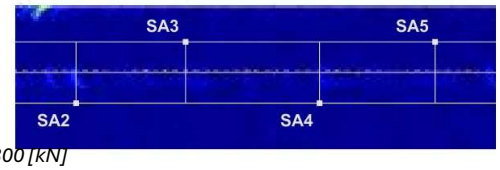
(d)-150 [kN]



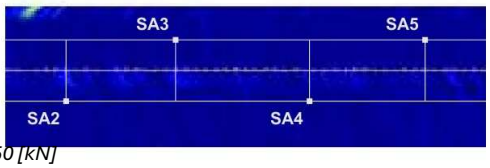
(e)-200 [kN]



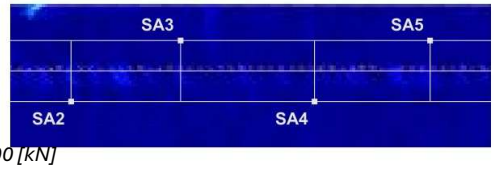
(f)-250 [kN]



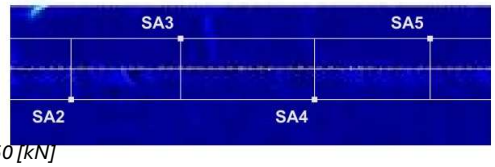
(g)-300 [kN]



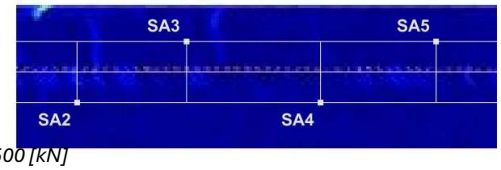
(h)-350 [kN]



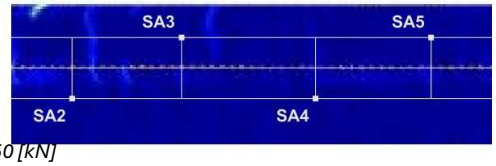
(i)-400 [kN]



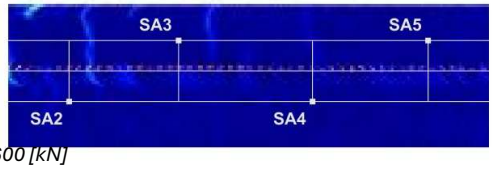
(j)-450 [kN]



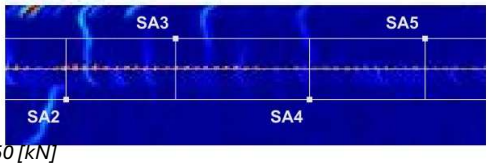
(k)-500 [kN]



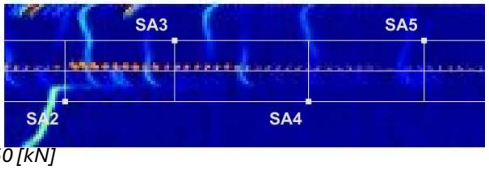
(l)-550 [kN]



(m)-600 [kN]



(n)-650 [kN]

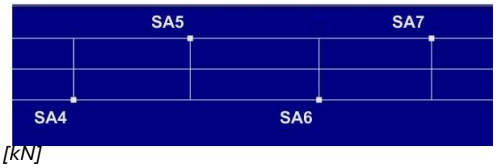


(o)-750 [kN]

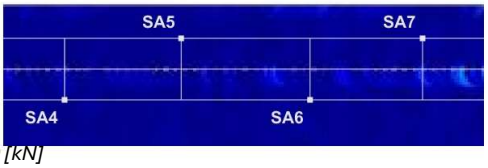
Group 2 – Overall Analysis (p1v2)						
Num. [-]	Load DIC [kN]	Load [kN]	Flexural Cracks		Shear Cracks	
1	0,1	0				
2	47,5	50				
3	100,7	100				
4	150,2	150				
5	198,0	200				
6	248,4	250				
7	299,8	300				
8	351,0	350				
9	401,2	400				
10	448,1	450	S, Fc F-VII			
11	500,9	500	S, Fc F-V			
12	549,7	550	Cs, Fc F-V			
13	598,2	600				
14	647,9	650	Cs, Sc F-VII	S, Fc F-V(2) &IX		
-	-	700	-	-	-	-
15	750,9	750	S, FC F- VII(2) &IX(2)			

Group 2 – Overall Analysis (p2v2)						
Num. [-]	Load DIC [kN]	Load [kN]	Delamination		Other Cracks	
1	0,1	0				
2	47,5	50				
3	100,7	100				
4	150,2	150				
5	198,0	200				
6	248,4	250				
7	299,8	300				
8	351,0	350				
9	401,2	400				
10	448,1	450				
11	500,9	500	S, D dt2			
12	549,7	550				
13	598,2	600				
14	647,9	650	Cs, D dt2	S, D dt3		
-	-	700	-	-	-	-
15	750,9	750	Cs, D dt3	S, D dt4		

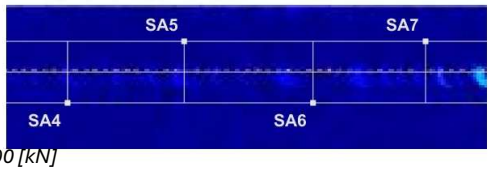
Group 3 – Group DIC Figures and Overall Analysis



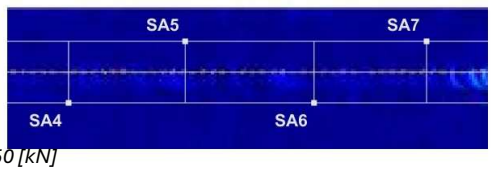
(a)-0 [kN]



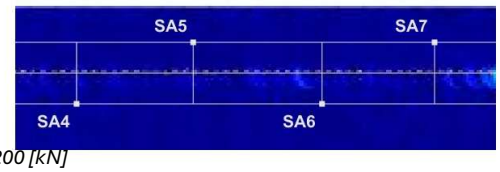
(b)-50 [kN]



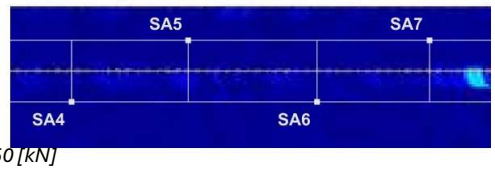
(c)-100 [kN]



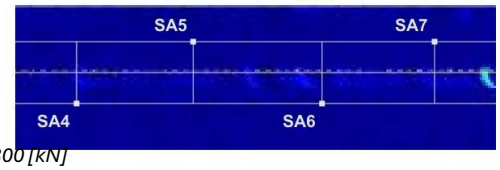
(d)-150 [kN]



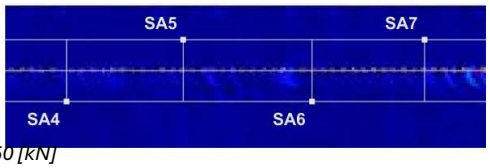
(e)-200 [kN]



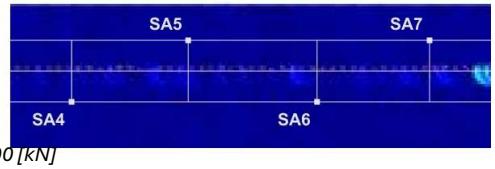
(f)-250 [kN]



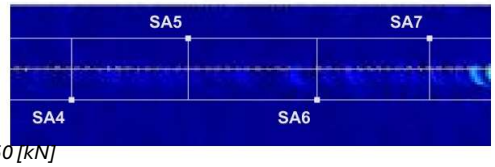
(g)-300 [kN]



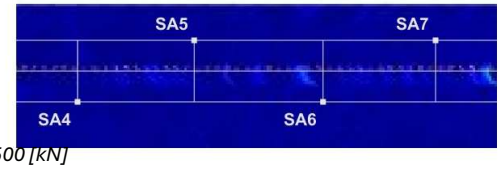
(h)-350 [kN]



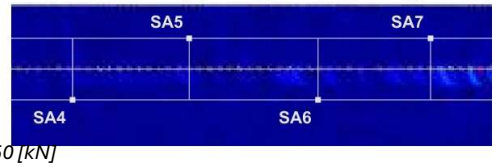
(i)-400 [kN]



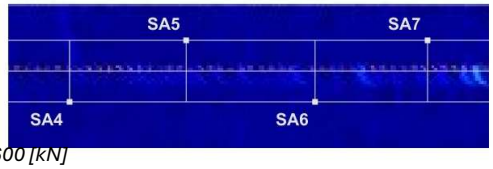
(j)-450 [kN]



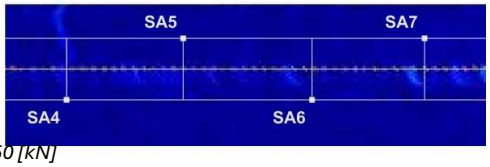
(k)-500 [kN]



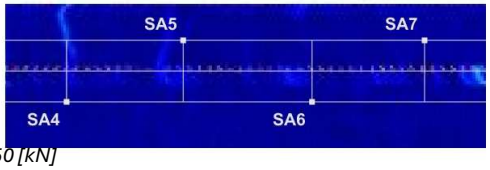
(l)-550 [kN]



(m)-600 [kN]



(n)-650 [kN]

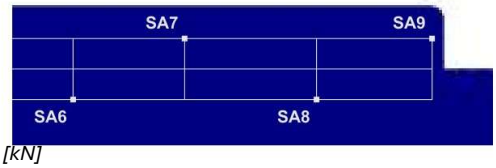


(o)-750 [kN]

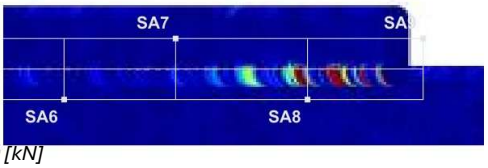
Group 3 – Overall Analysis (p1v2)						
Num. [-]	Load DIC [kN]	Load [kN]	Flexural Cracks		Shear Cracks	
1	0,1	0				
2	47,5	50				
3	100,7	100				
4	150,2	150				
5	198,0	200				
6	248,4	250				
7	299,8	300				
8	351,0	350				
9	401,2	400				
10	448,1	450				
11	500,9	500				
12	549,7	550				
13	598,2	600				
14	647,9	650	Cs, Sc F-VII	S, Fc F-IX		
-	-	700	-	-	-	-
15	750,9	750	S, FC F-IX(2)			

Group 3 – Overall Analysis (p2v2)						
Num. [-]	Load DIC [kN]	Load [kN]	Delamination		Other Cracks	
1	0,1	0				
2	47,5	50				
3	100,7	100				
4	150,2	150				
5	198,0	200				
6	248,4	250				
7	299,8	300				
8	351,0	350				
9	401,2	400				
10	448,1	450				
11	500,9	500				
12	549,7	550				
13	598,2	600				
14	647,9	650				
-	-	700	-	-	-	-
15	750,9	750				

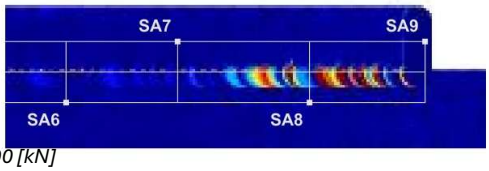
Group 4 – Group DIC Figures and Overall Analysis



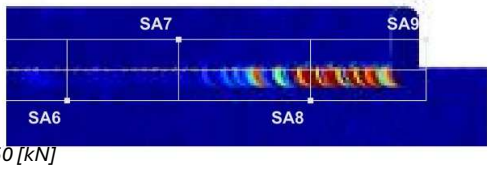
(a)-0 [kN]



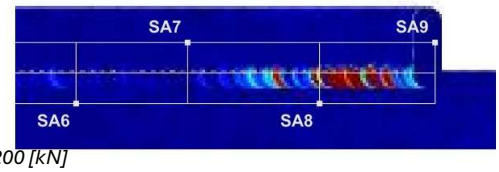
(b)-50 [kN]



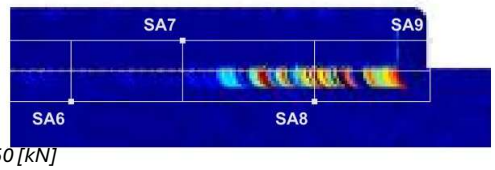
(c)-100 [kN]



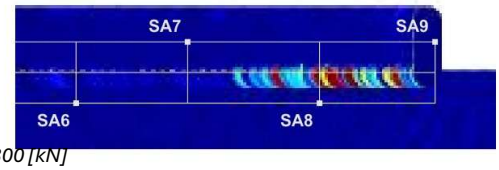
(d)-150 [kN]



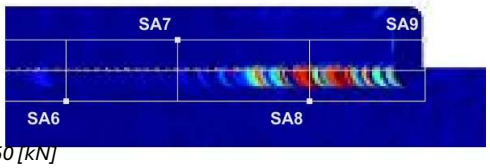
(e)-200 [kN]



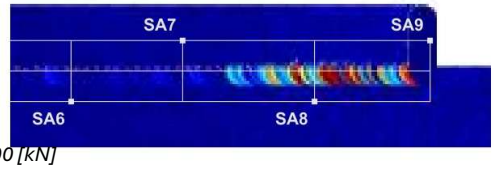
(f)-250 [kN]



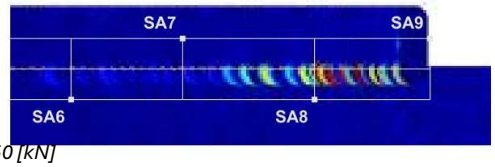
(g)-300 [kN]



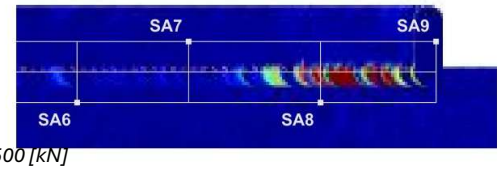
(h)-350 [kN]



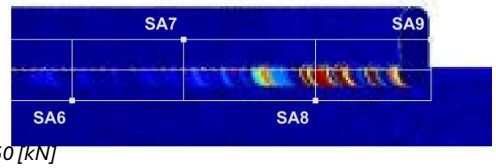
(i)-400 [kN]



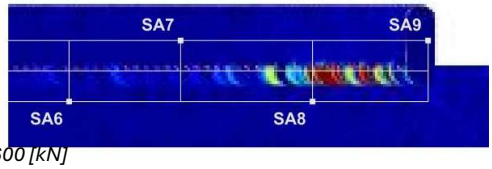
(j)-450 [kN]



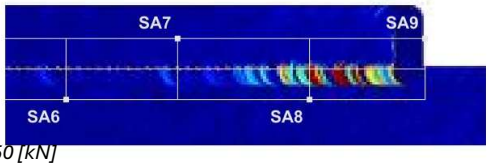
(k)-500 [kN]



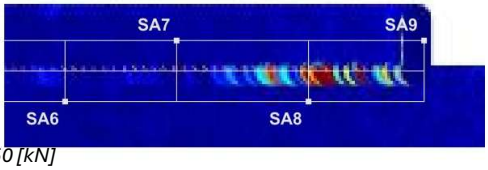
(l)-550 [kN]



(m)-600 [kN]



(n)-650 [kN]



(o)-750 [kN]

Group 4 – Overall Analysis (p1v2)						
Num. [-]	Load DIC [kN]	Load [kN]	Flexural Cracks		Shear Cracks	
1	0,1	0				
2	47,5	50				
3	100,7	100				
4	150,2	150				
5	198,0	200				
6	248,4	250				
7	299,8	300				
8	351,0	350				
9	401,2	400				
10	448,1	450				
11	500,9	500				
12	549,7	550				
13	598,2	600				
14	647,9	650				
-	-	700	-	-	-	-
15	750,9	750				

Group 4 – Overall Analysis (p2v2)						
Num. [-]	Load DIC [kN]	Load [kN]	Delamination		Other Cracks	
1	0,1	0				
2	47,5	50				
3	100,7	100				
4	150,2	150				
5	198,0	200				
6	248,4	250				
7	299,8	300				
8	351,0	350				
9	401,2	400				
10	448,1	450				
11	500,9	500				
12	549,7	550				
13	598,2	600				
14	647,9	650				
-	-	700	-	-	-	-
15	750,9	750				

Appendix V – Ultrasonic Graphs Analysis

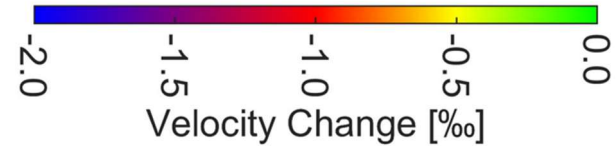
Appendix III is organized as follows:

- **Appendix IV.1** – Ultrasonic Graphs of beam S10H2D
- **Appendix IV.2** – Ultrasonic Graphs of beam S10H2A
- **Appendix IV.3** – Ultrasonic Graphs of beam S10H1D

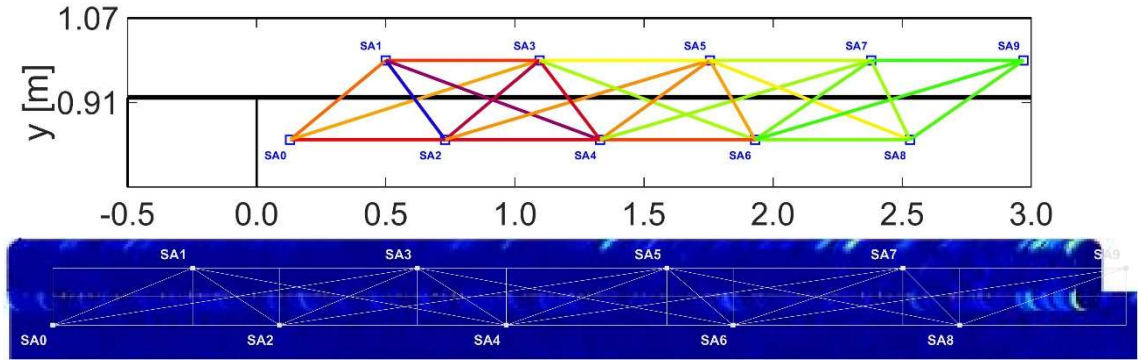
Note: Each sub-chapter as for all load available the four Ultra-sonic Graph to DIC observations

Appendix V.1 – beam S10H2D

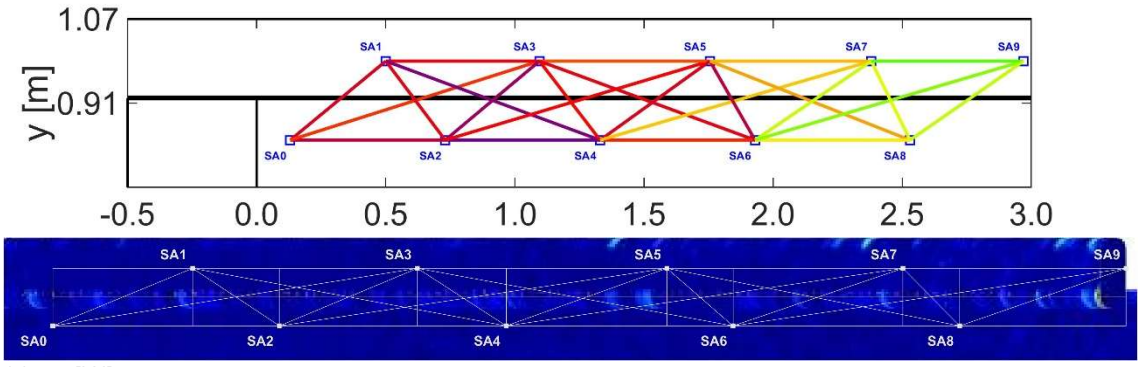
CWI – Relative Velocity Change (ε) vs DIC



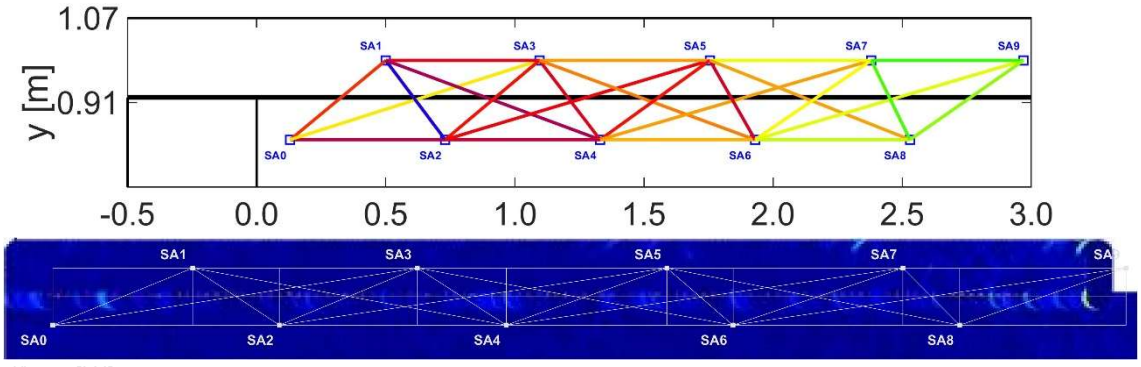
(b)-(p)



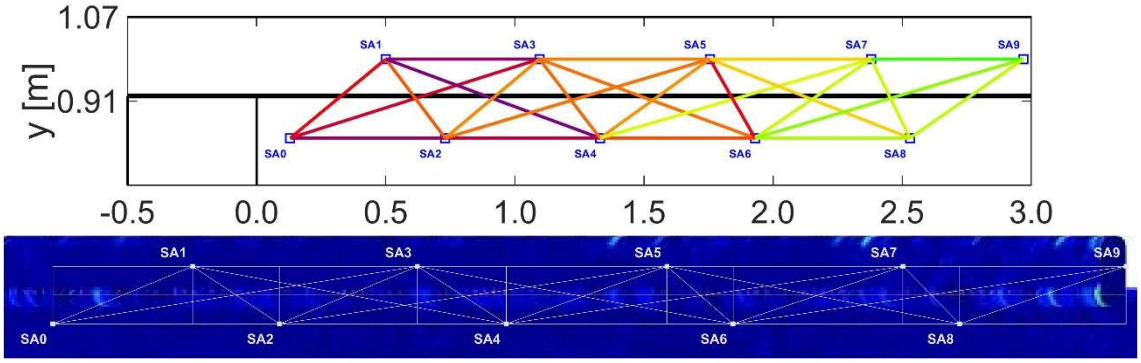
(b)- 50[kN]



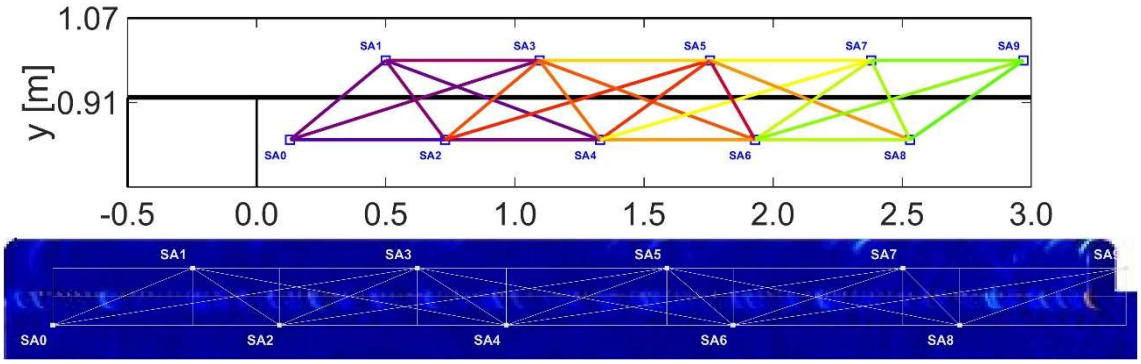
(c)- 100[kN]



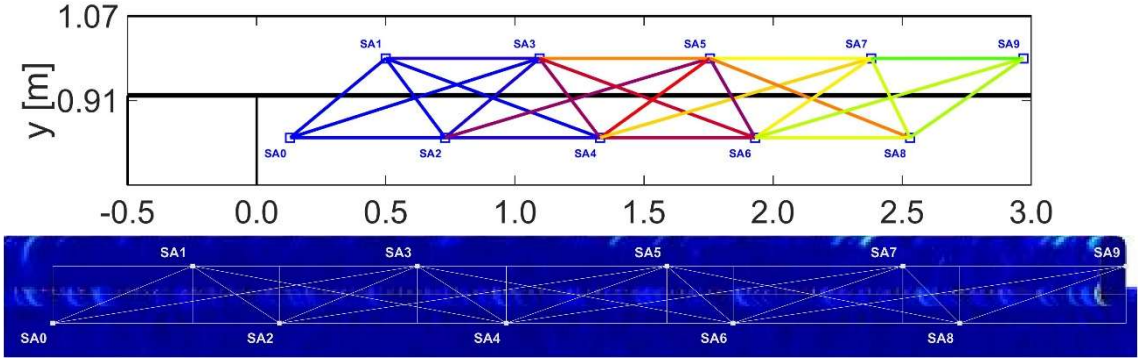
d()- 150[kN]



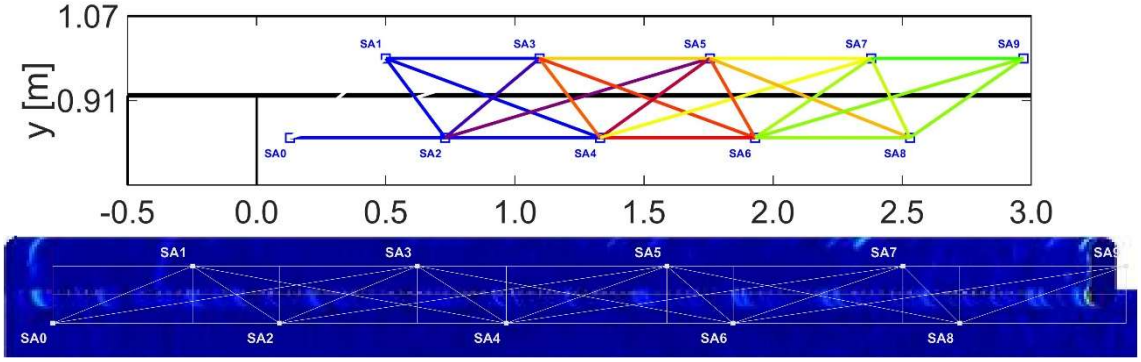
(e)- 200[kN]



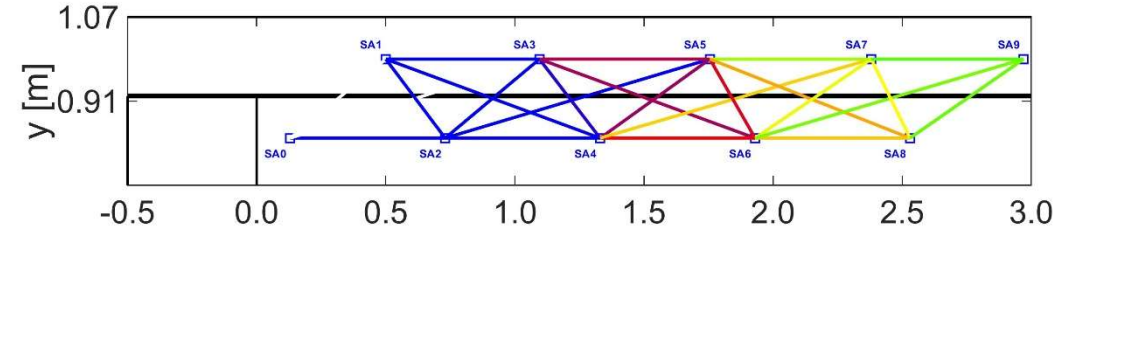
(f)- 250[kN]

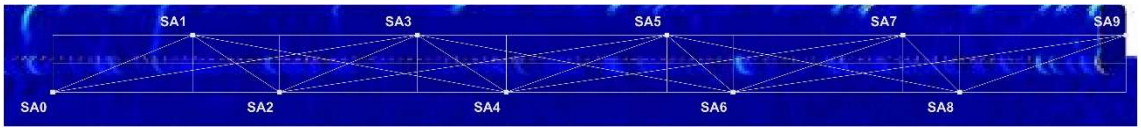


(g)- 300[kN]

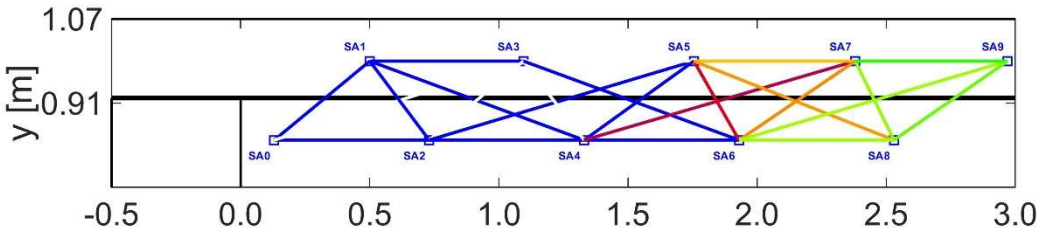


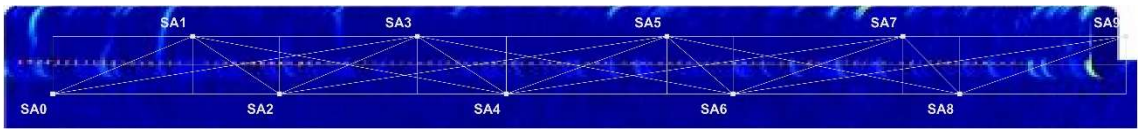
(h)- 350[kN]



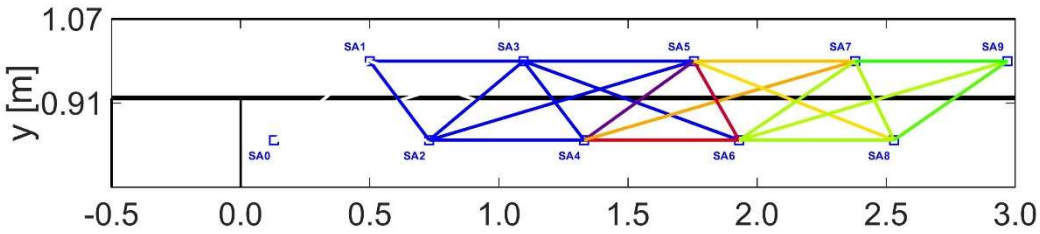


(i)- 400[kN]



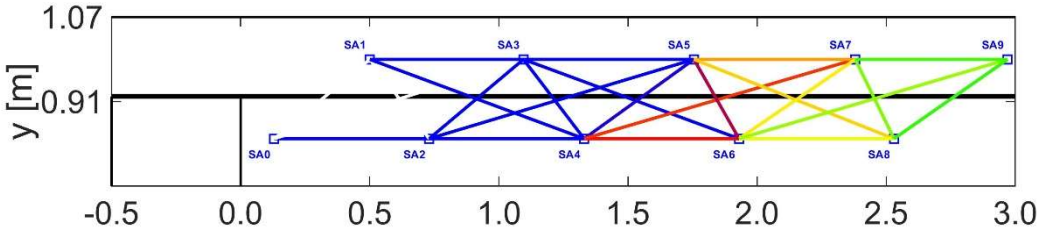


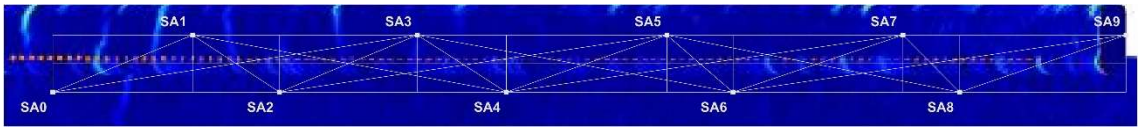
(j)- 450[kN]



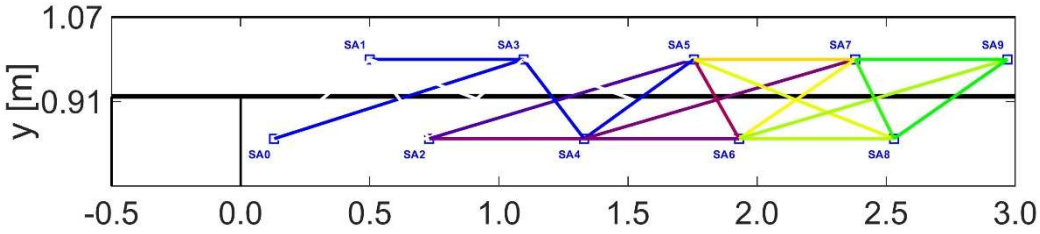
(no DIC Figure)

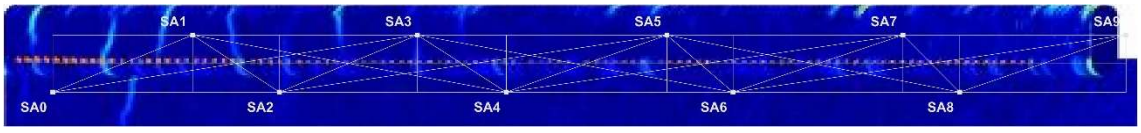
(k)- 500[kN]



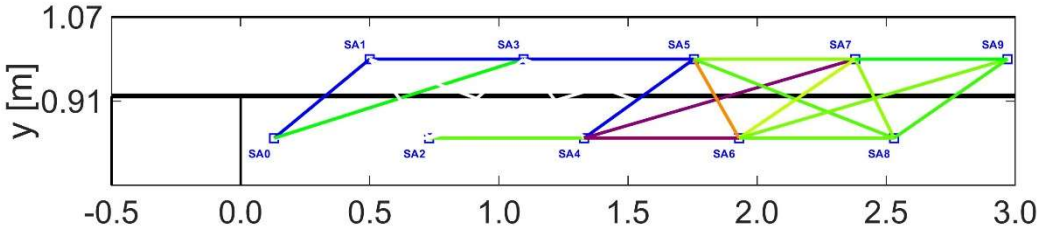


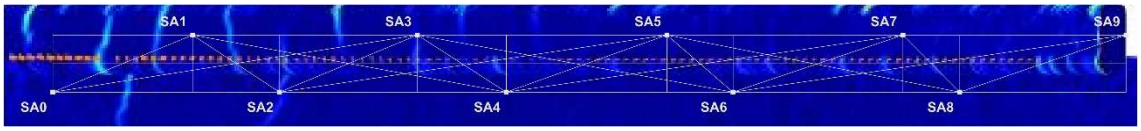
(l)- 550[kN]



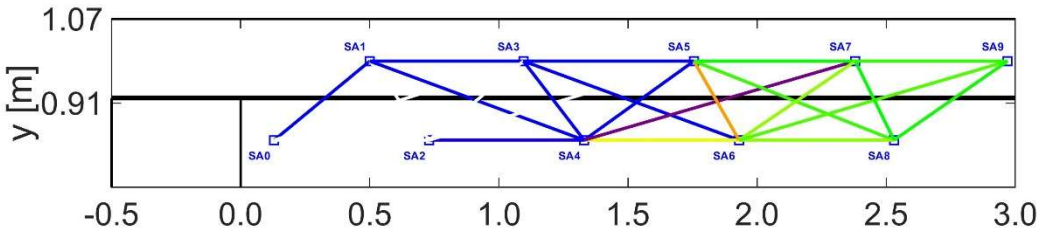


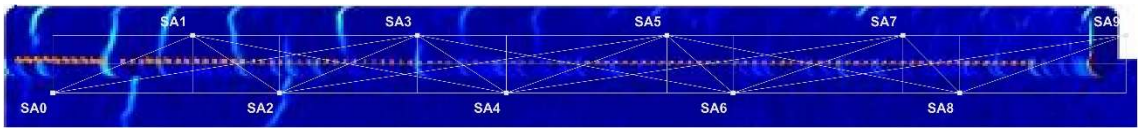
(m)- 600[kN]



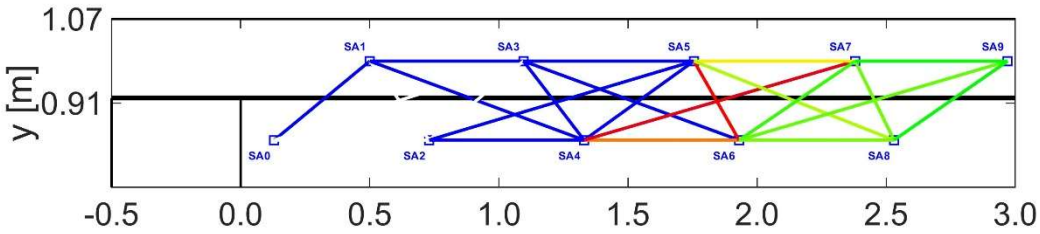


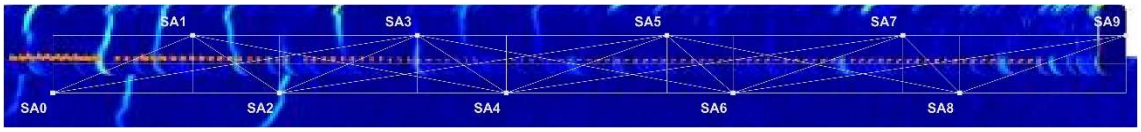
(n)- 650[kN]





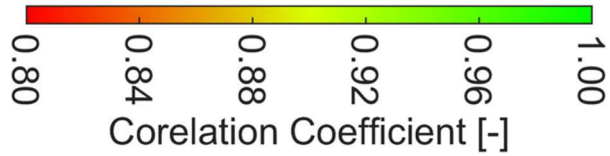
(o)- 700[kN]



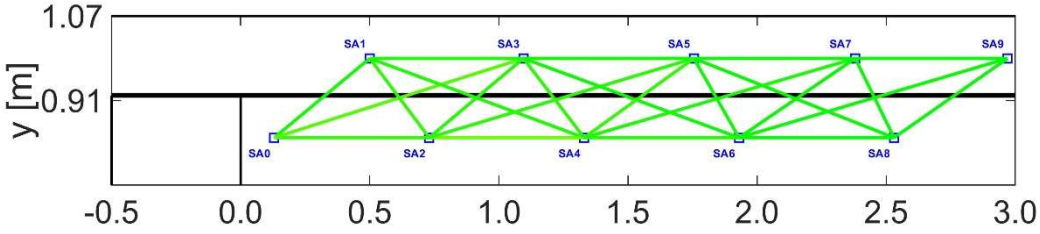


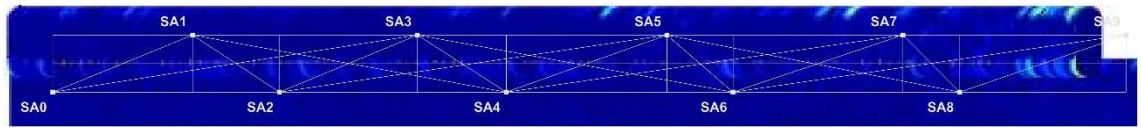
(p)- 750[kN]

CWI – Correlation Coefficient (CC) vs DIC

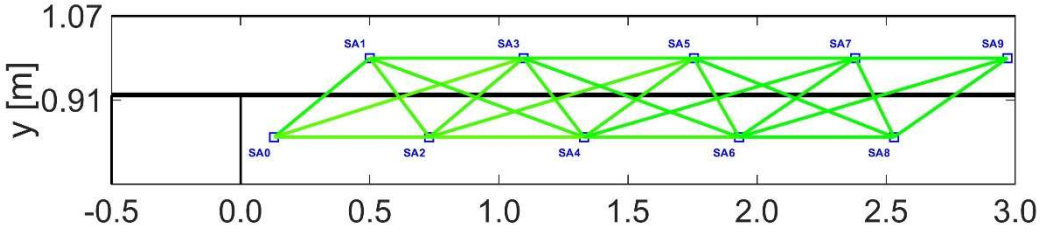


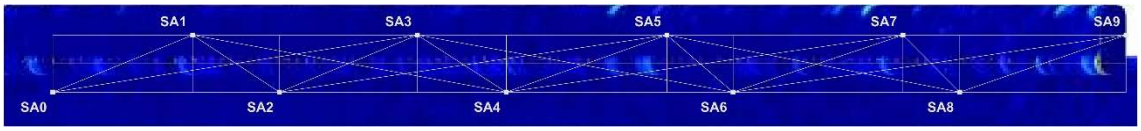
(b)-(p)



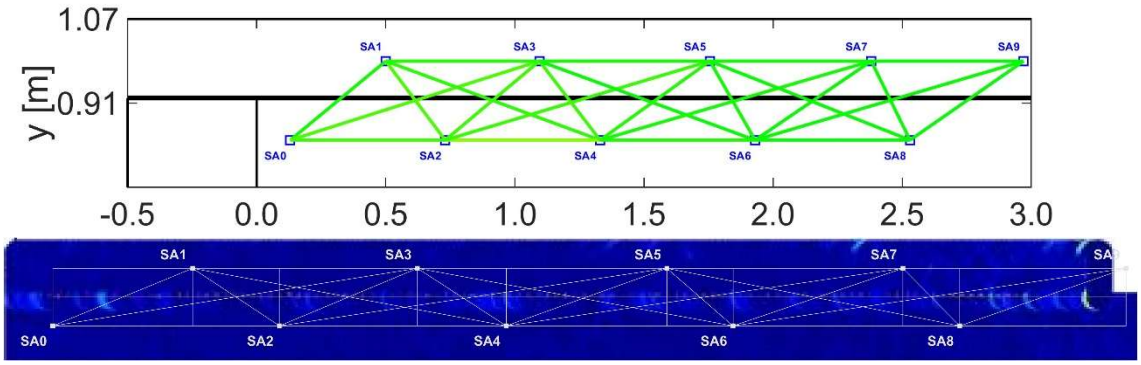


(b)- 50[kN]

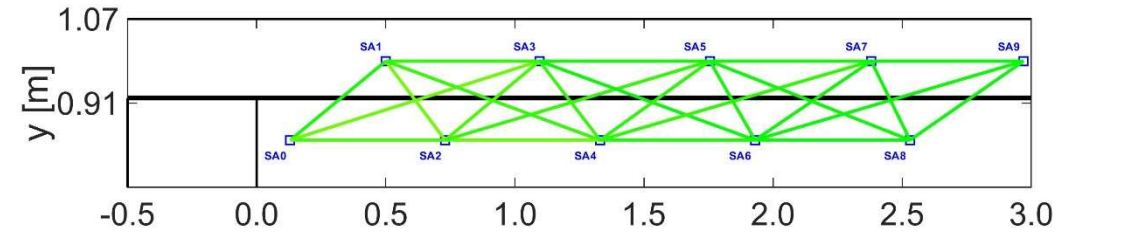


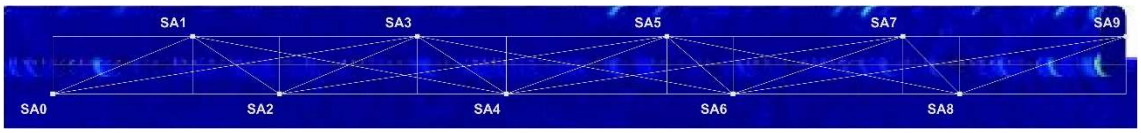


(c)- 100[kN]

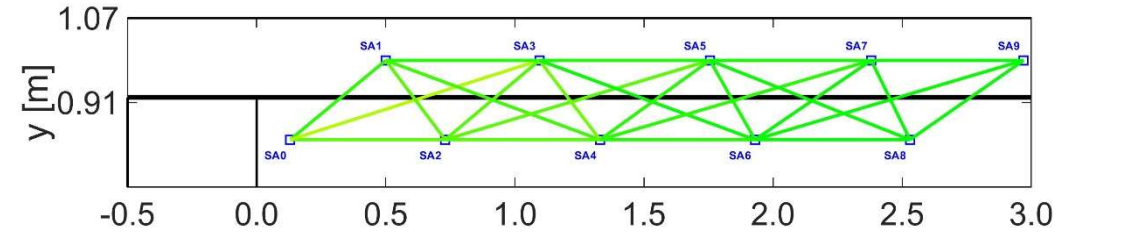


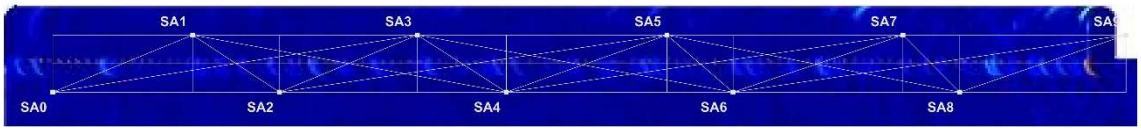
(d)- 150[kN]



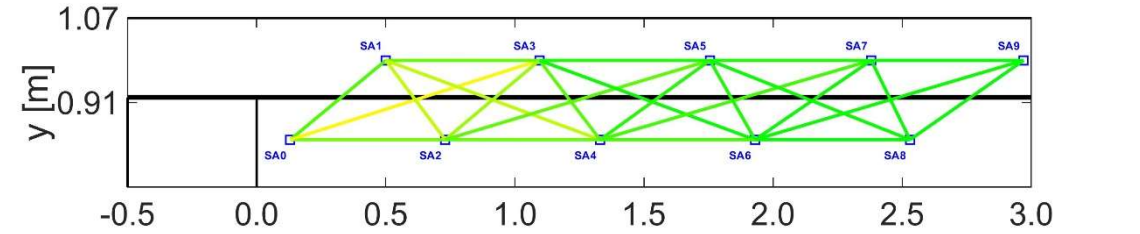


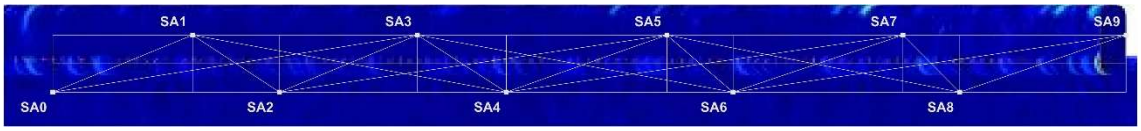
(e)- 200[kN]



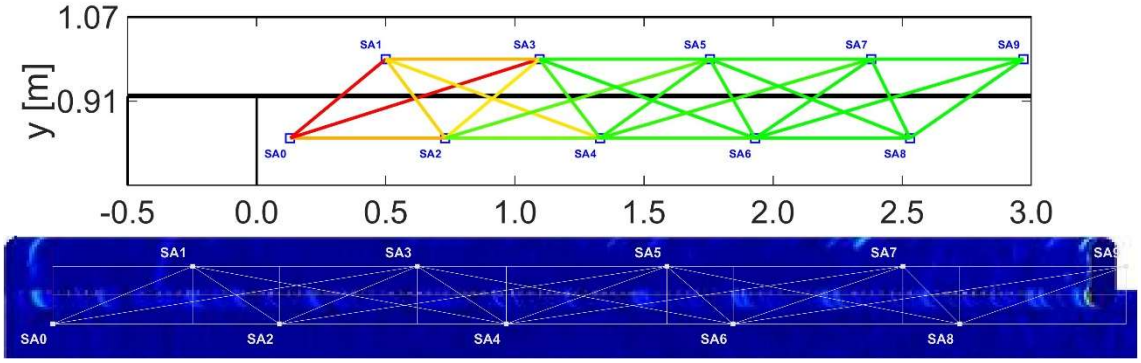


(f)- 250[kN]

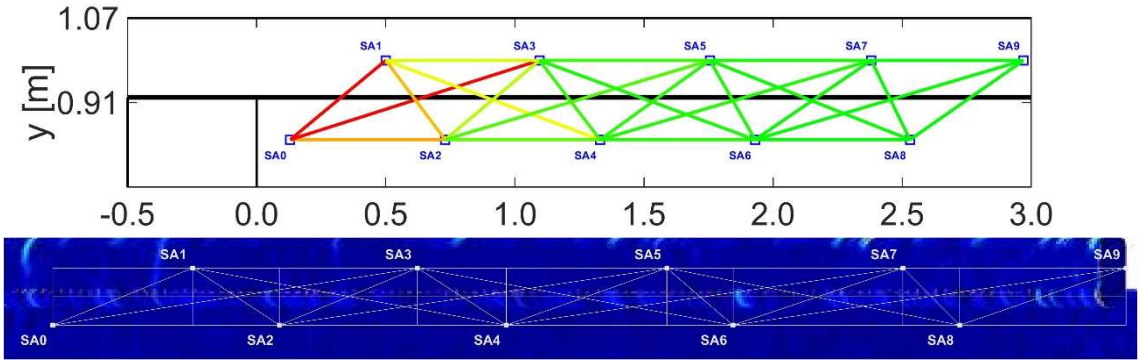




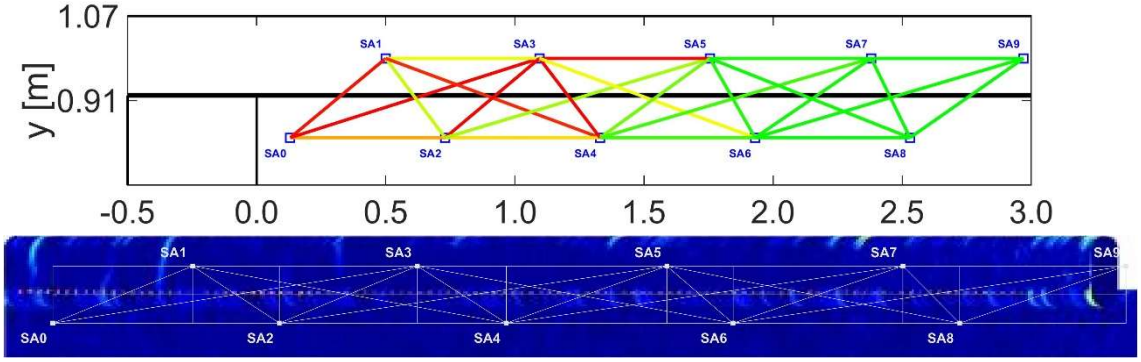
(g)- 300[kN]



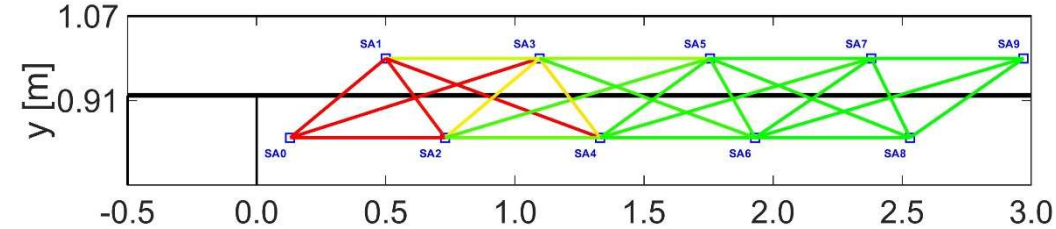
(h)- 350[kN]



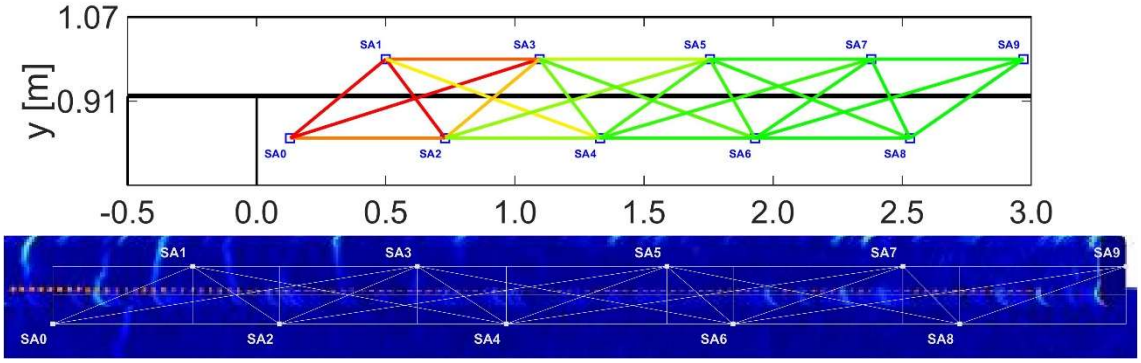
(i)- 400[kN]



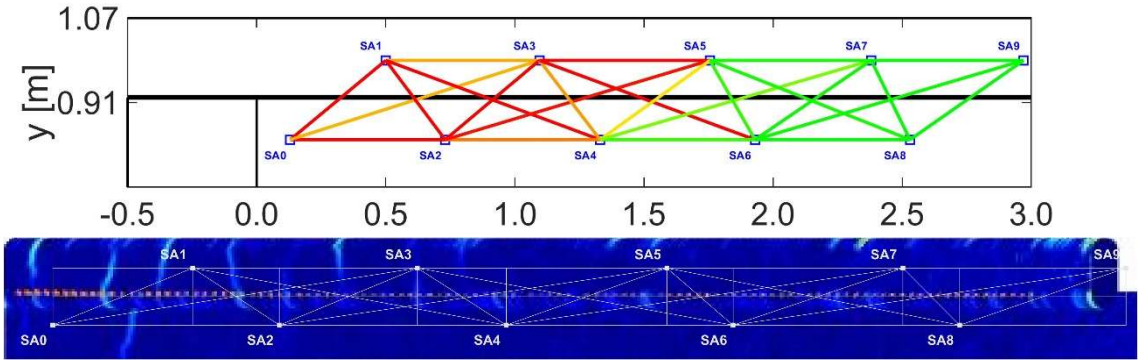
(j)- 450[kN]



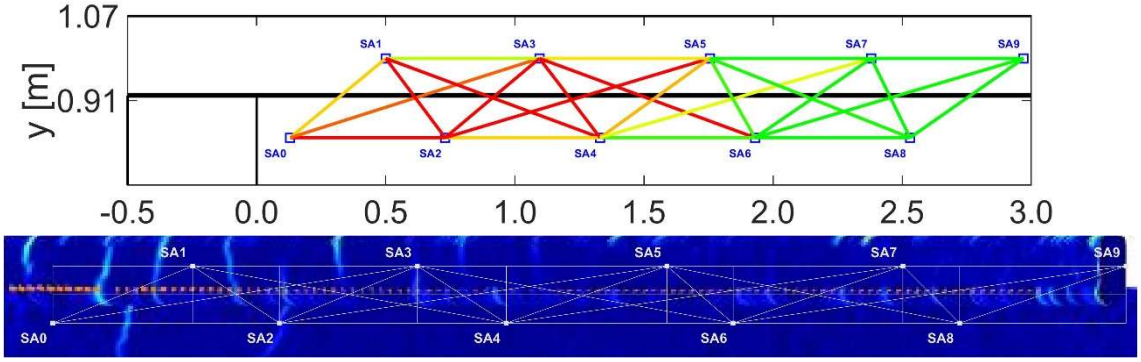
(no DIC Figure)
(k)- 500[kN]



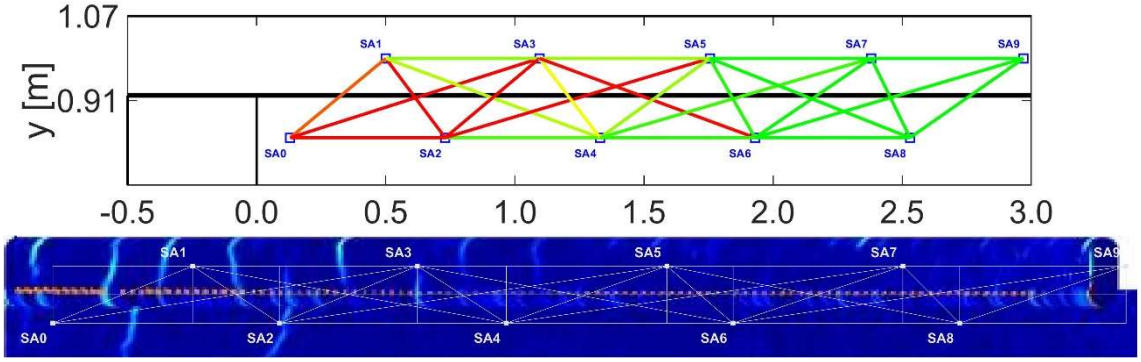
(l)- 550[kN]



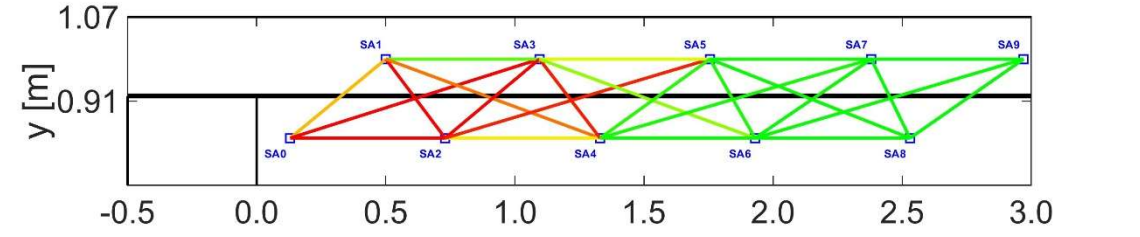
(m)- 600[kN]

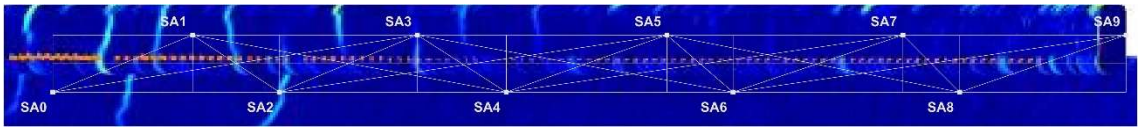


(n)- 650[kN]



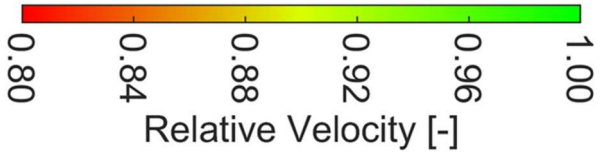
(o)- 700[kN]



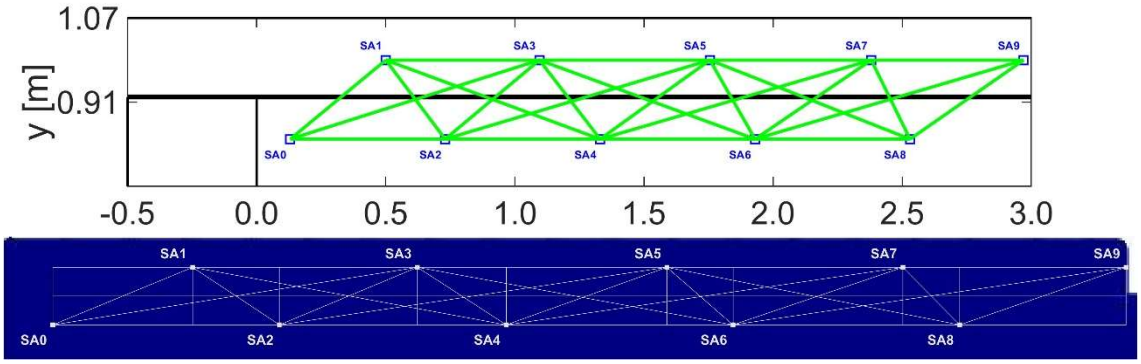


(p)- 750[kN]

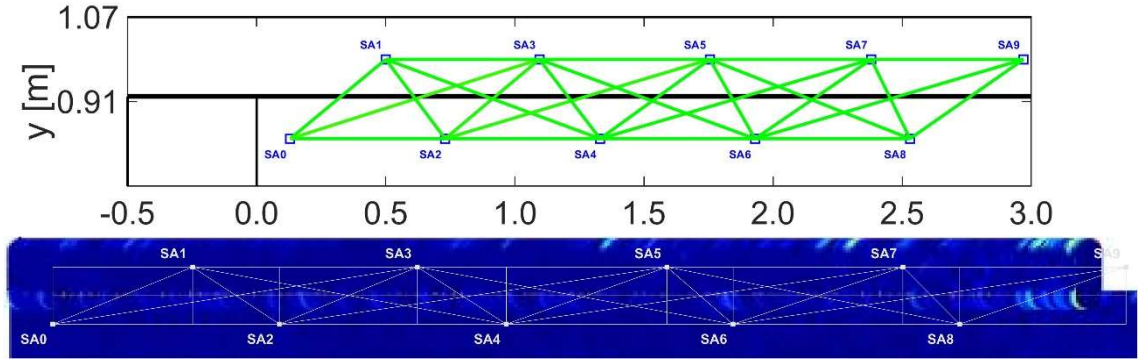
UPV – Relative Velocity vs DIC



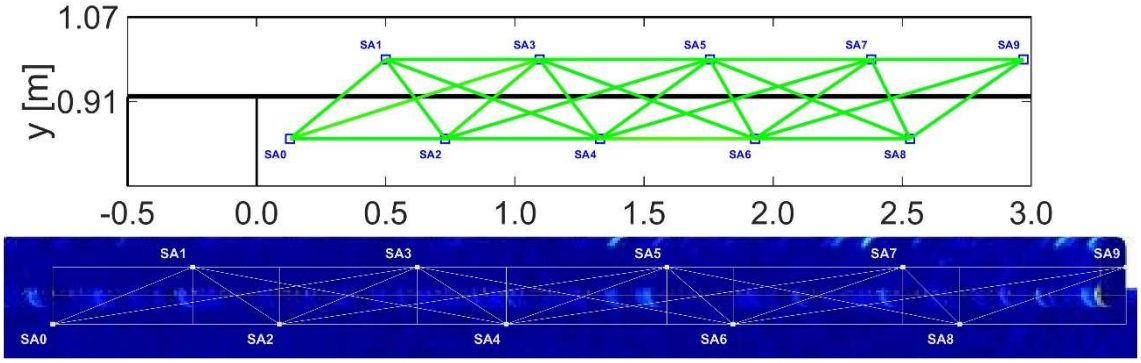
()-(ag)



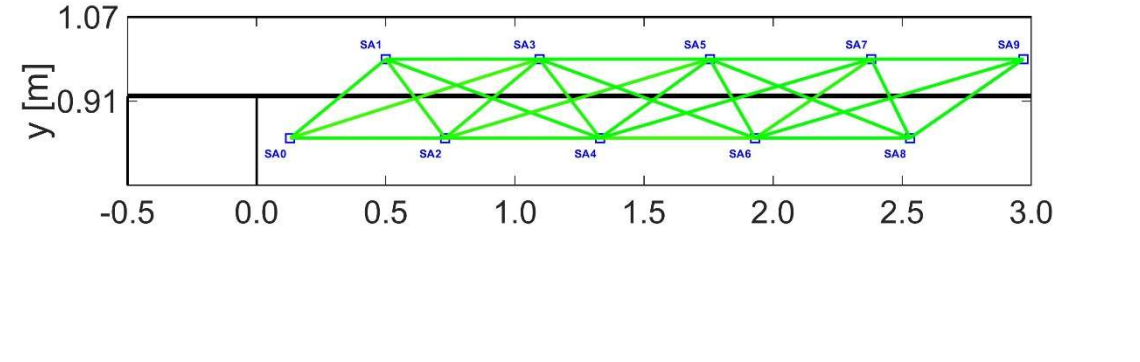
(b)- 0[kN]

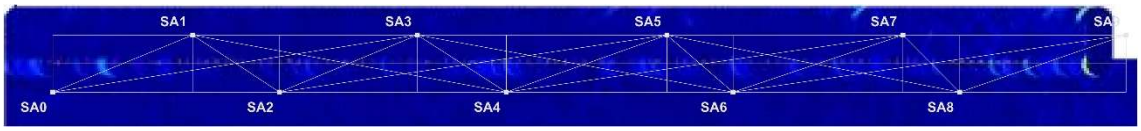


(b)- 50[kN]

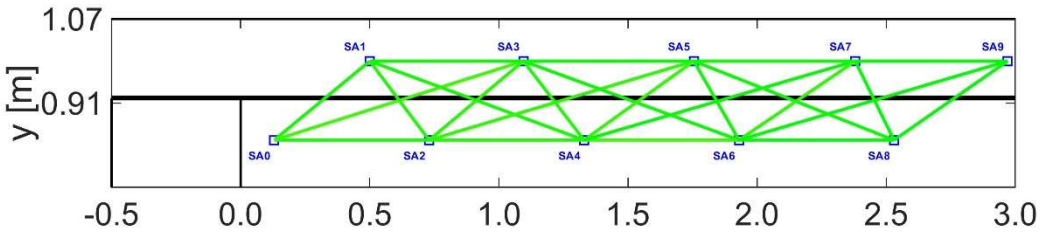


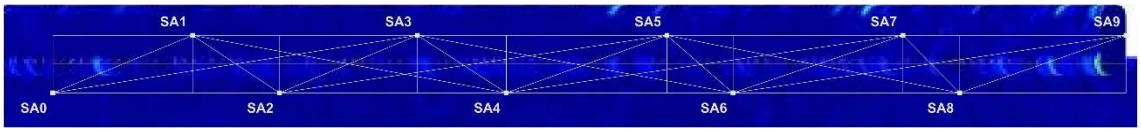
(c)- 100[kN]



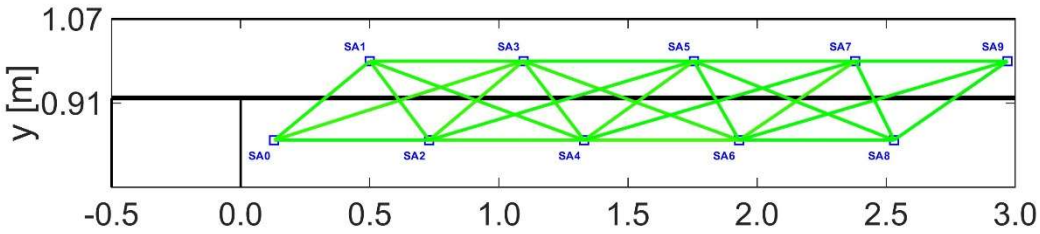


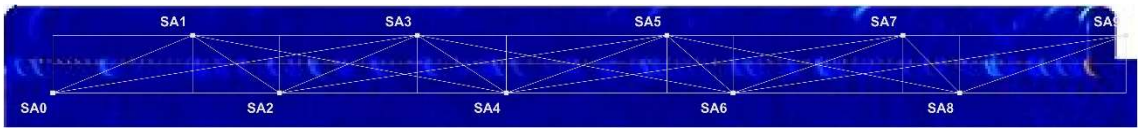
(d)- 150[kN]



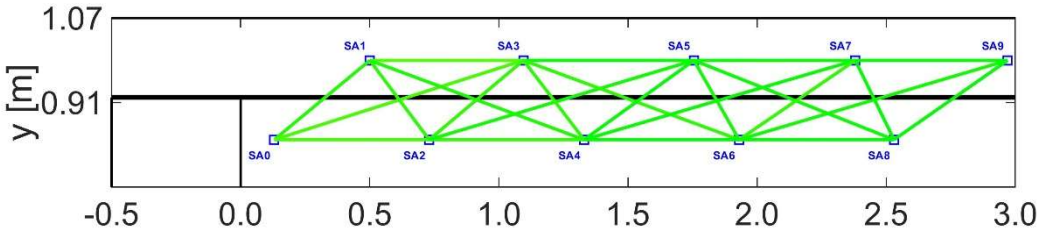


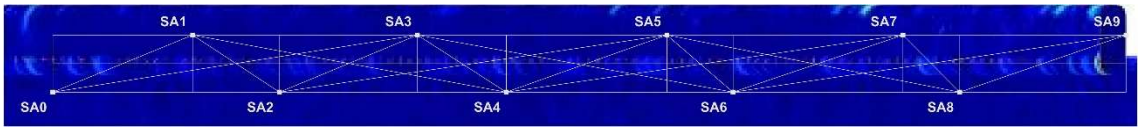
(e)- 200[kN]



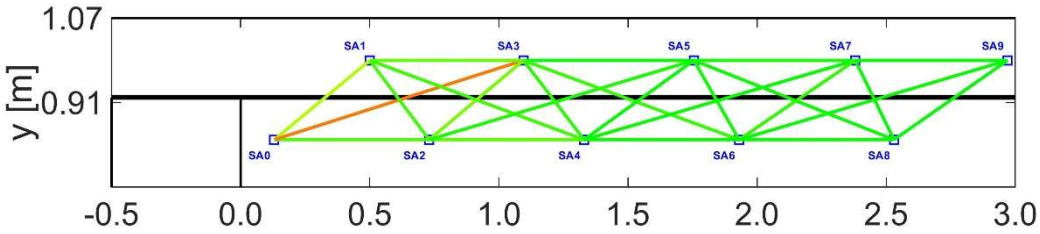


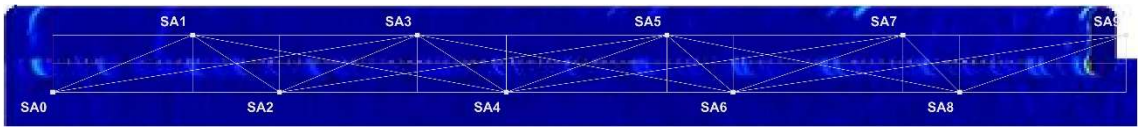
(f)- 250[kN]



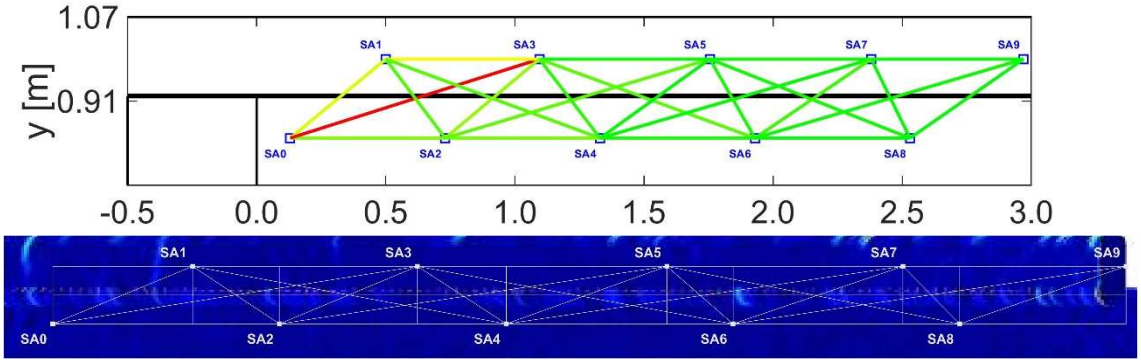


(g)- 300[kN]

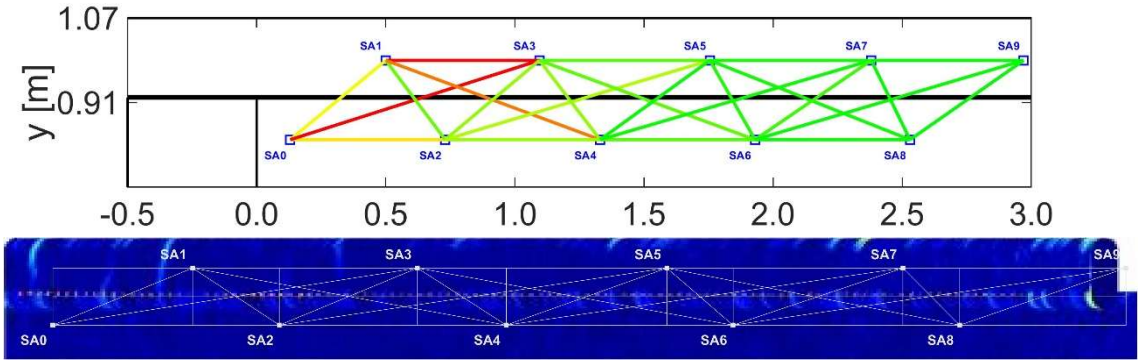




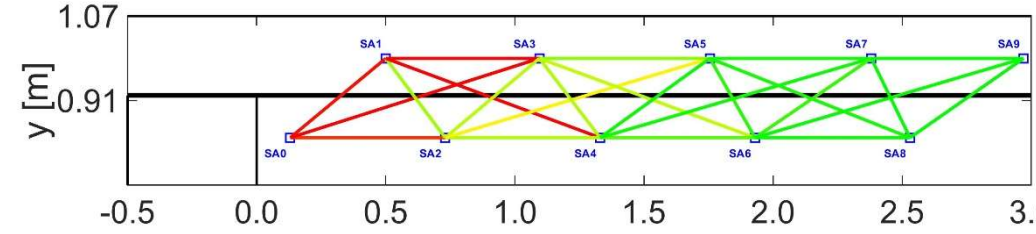
(h)- 350[kN]



(i)- 400[kN]

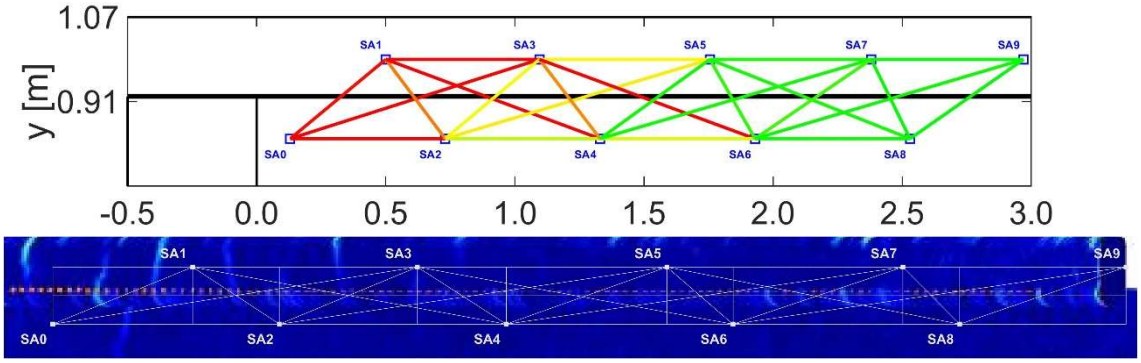


(j)- 450[kN]

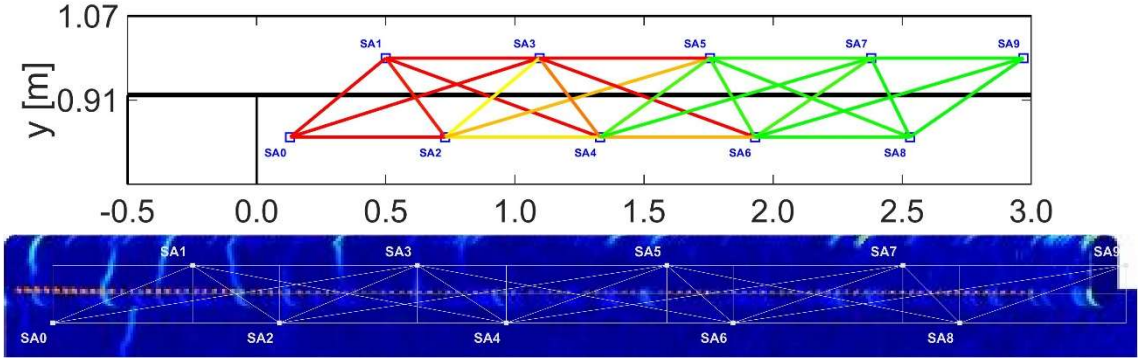


(no DIC Figure)

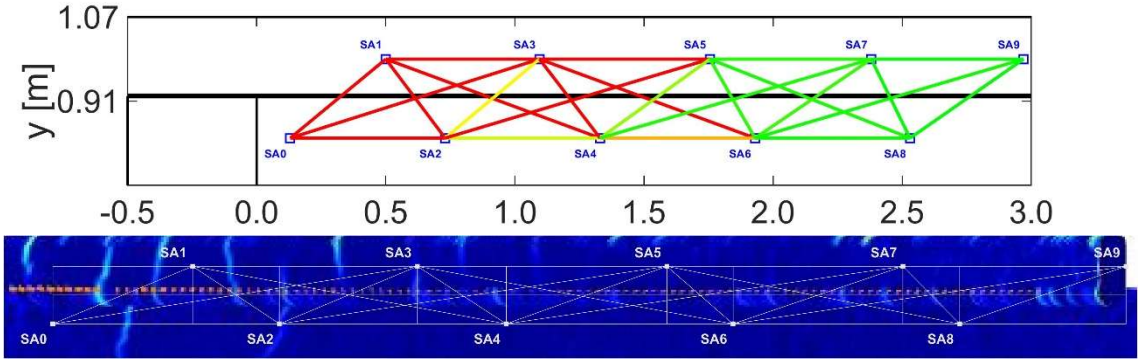
(k)- 500[kN]



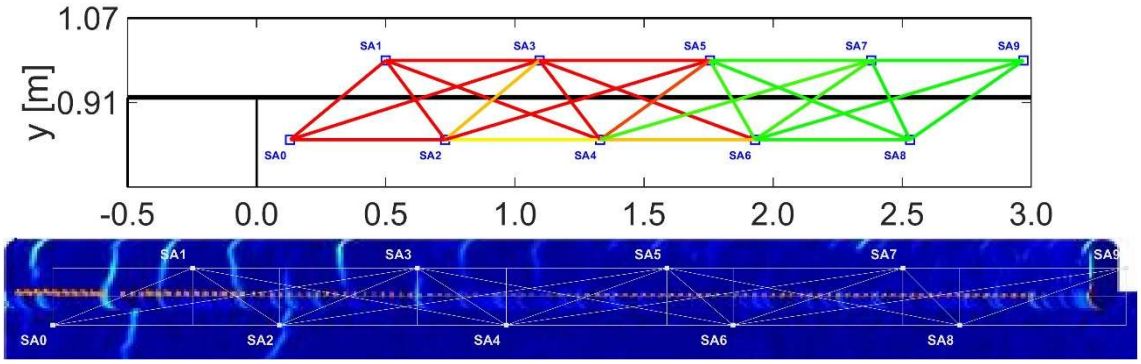
(l)- 550[kN]



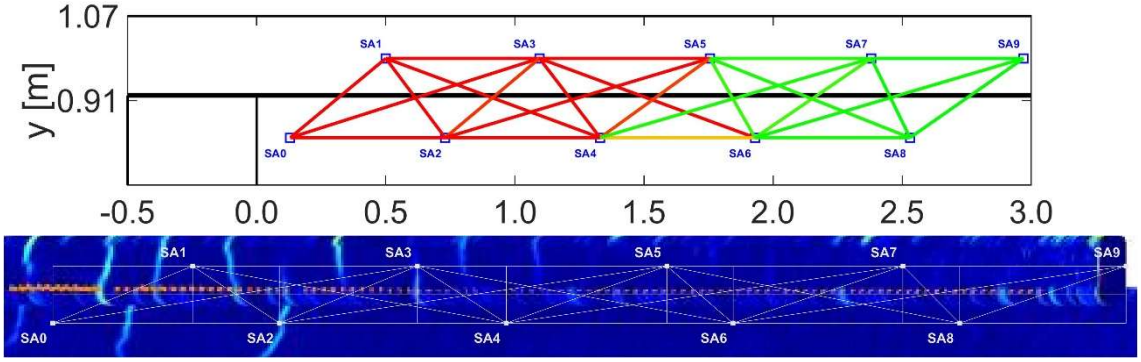
(m)- 600[kN]



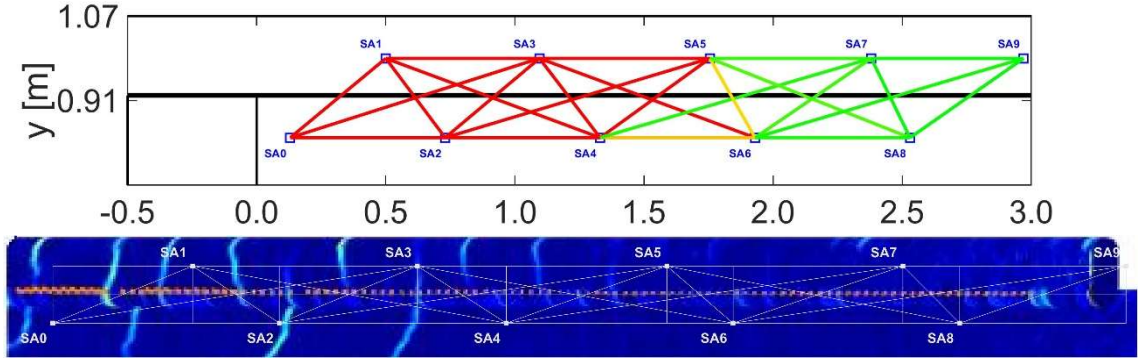
(n)- 650[kN]



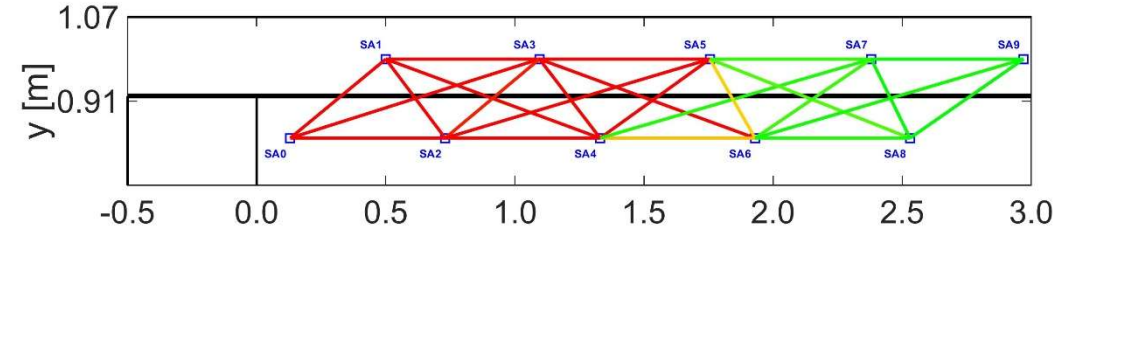
(o)- 700[kN]

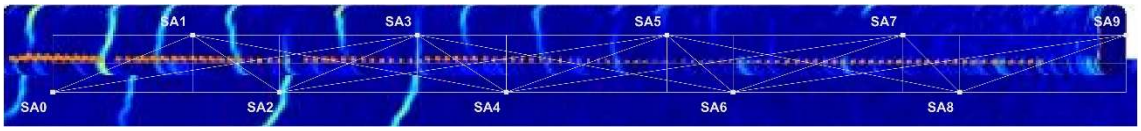


(p)- 750[kN]

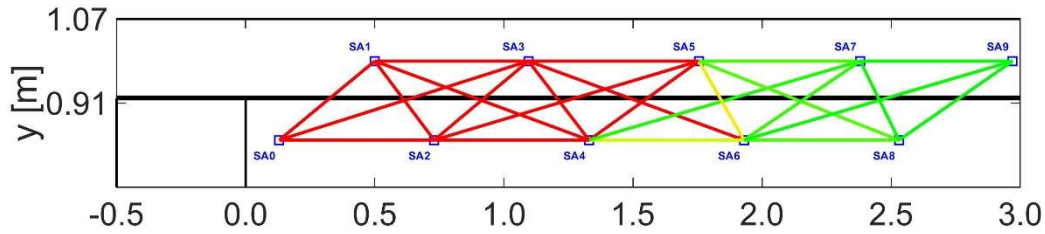


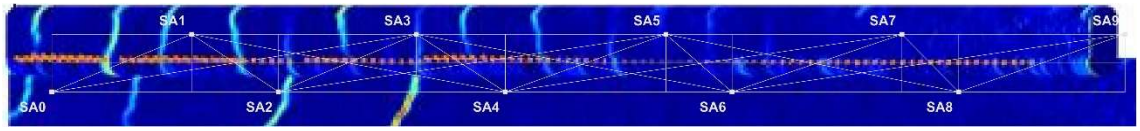
(q)- 800[kN]



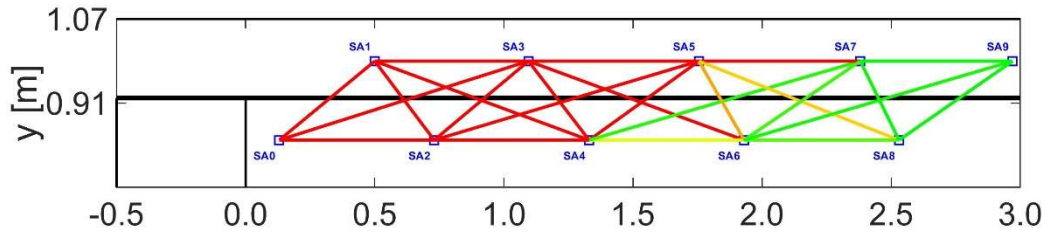


(r)- 850[kN]



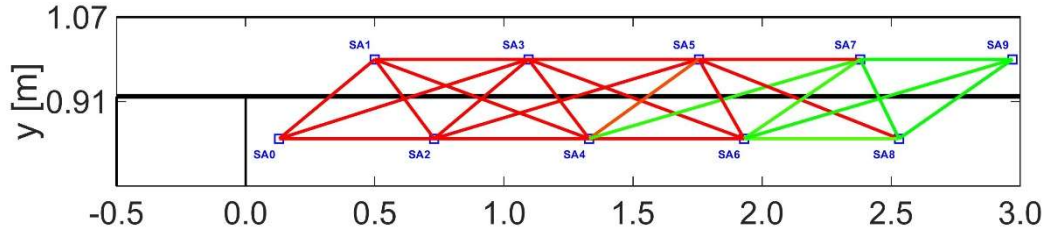


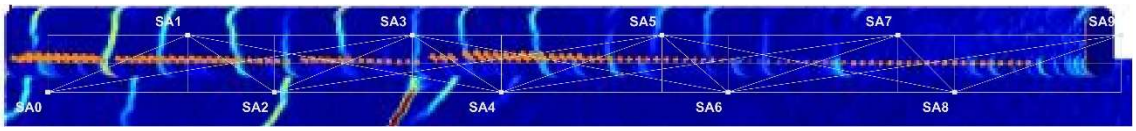
(s)- 900[kN]



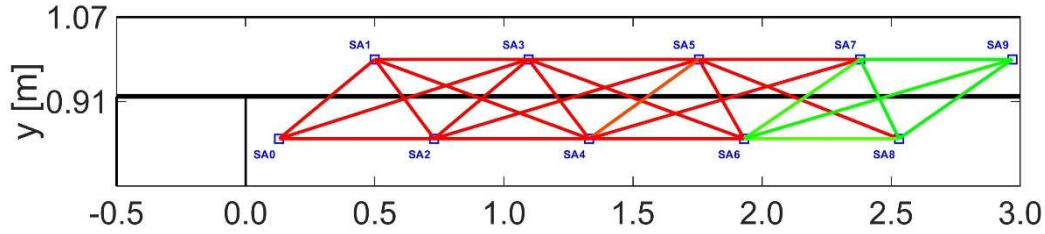
(no DIC Figure)

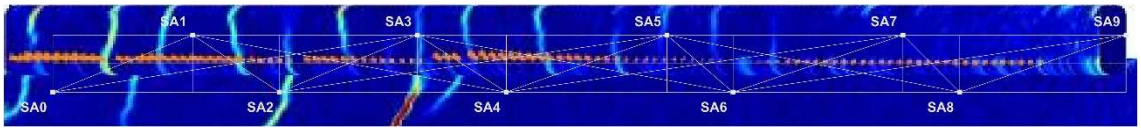
(t)- 950[kN]



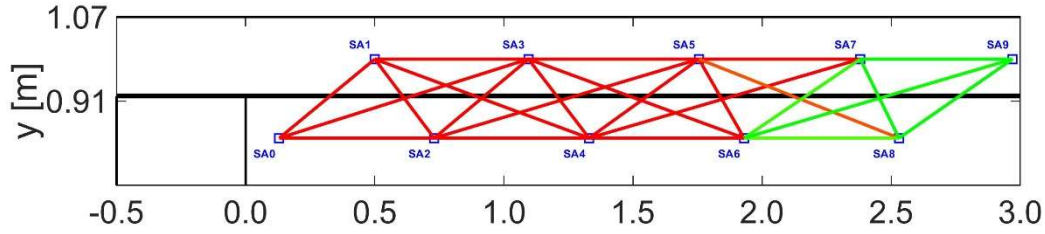


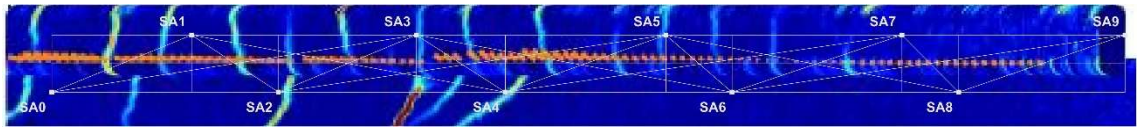
(u)- 1000[kN]



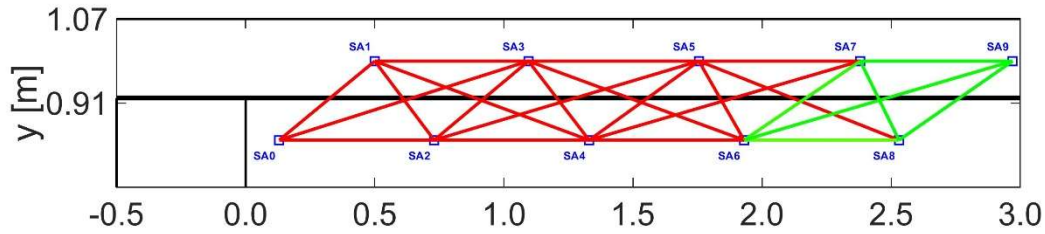


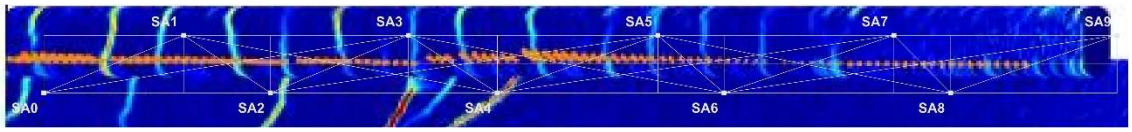
(v)- 1050[kN]



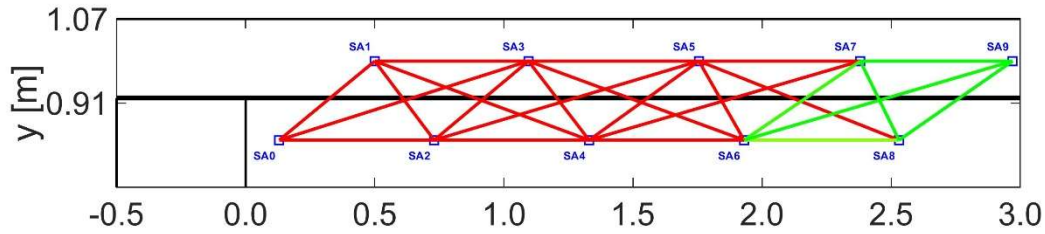


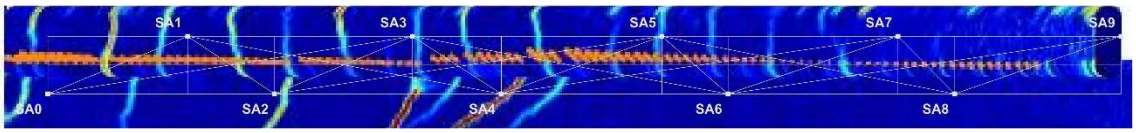
(w)- 1100[kN]



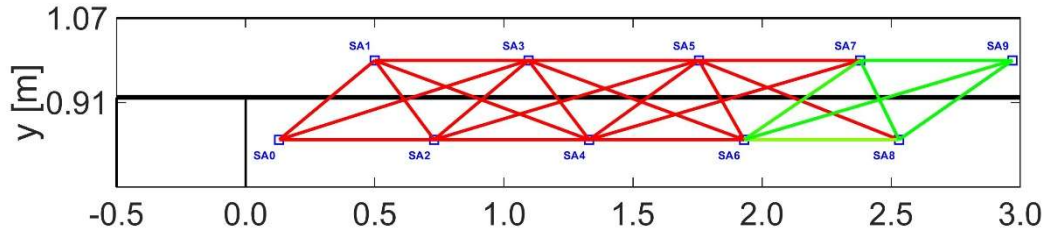


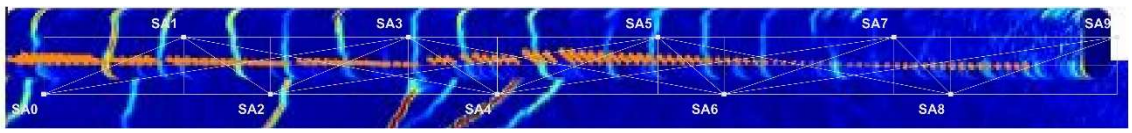
(x)- 1150[kN]



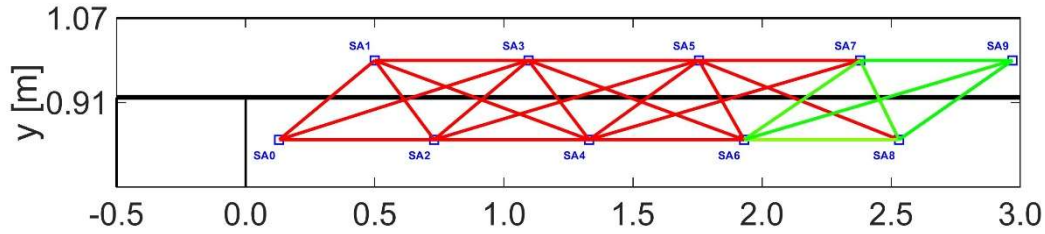


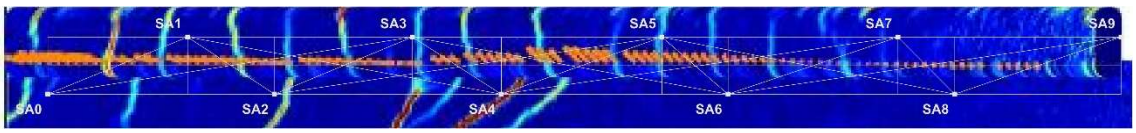
(y)- 1200[kN]



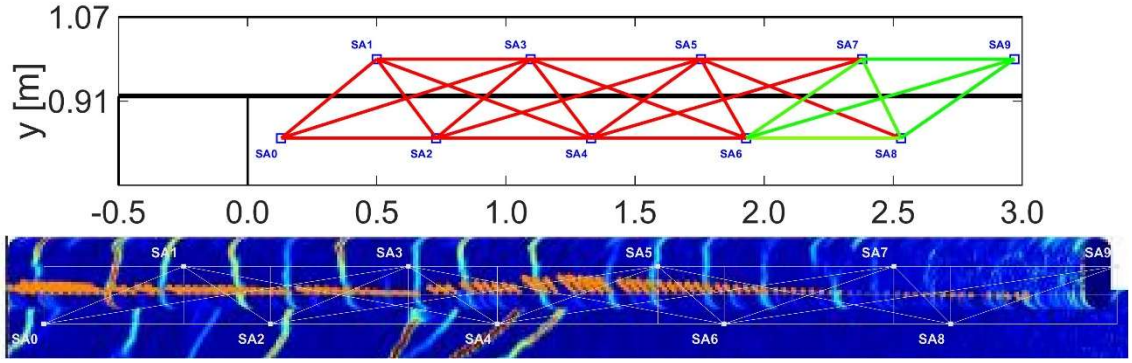


(z)- 1250[kN]

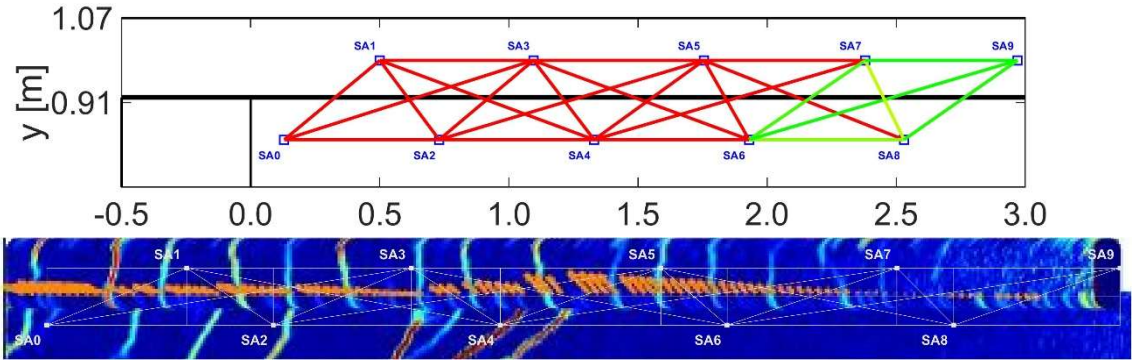




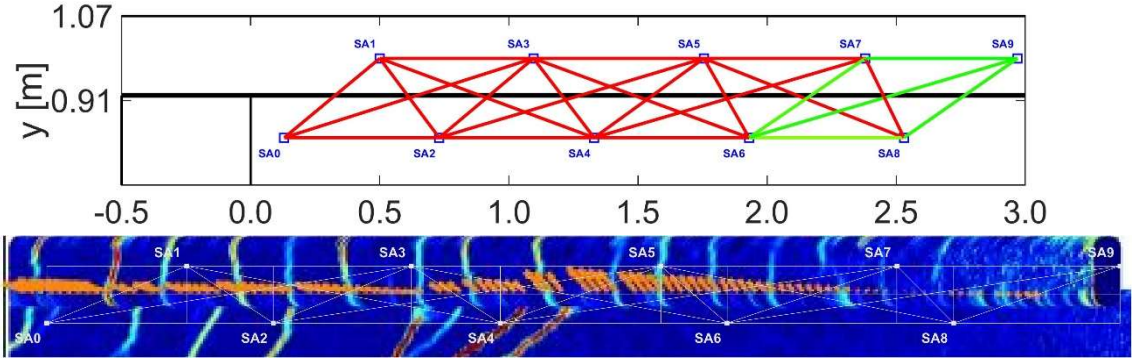
(aa)- 1300[kN]



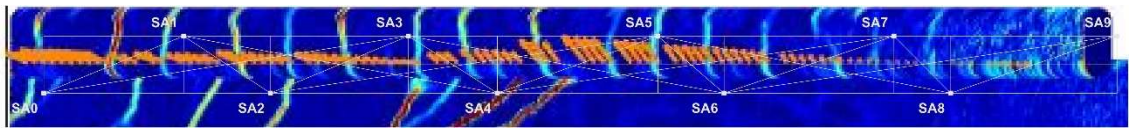
(ab)- 1350[kN]



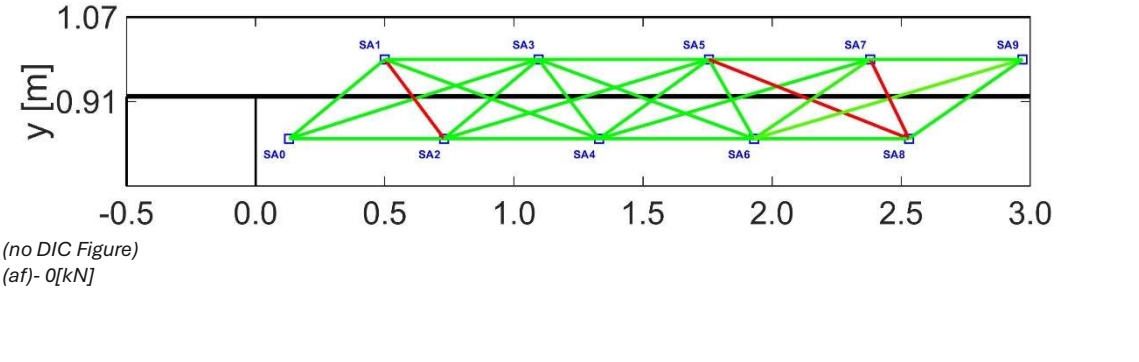
(ac)- 1400[kN]



(ad)- 1450[kN]

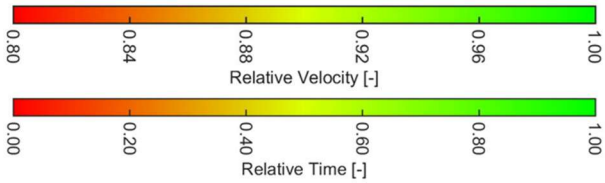


(ae)- 1500[kN]

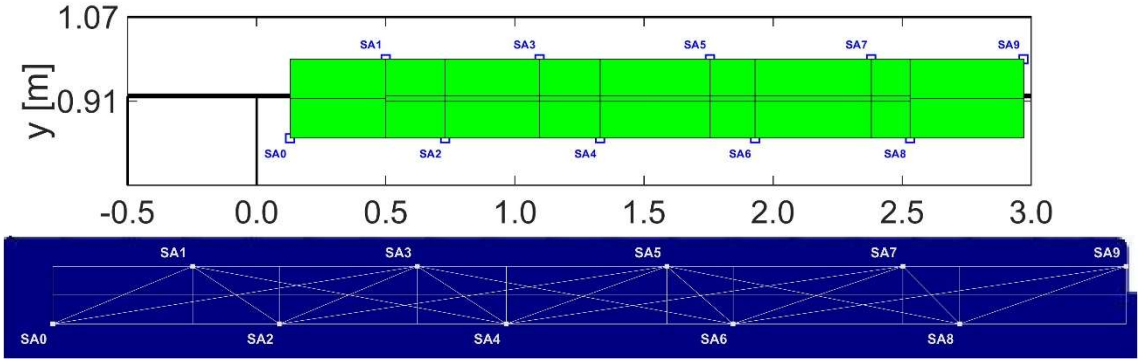


(no DIC Figure)
(af)- 0[kN]

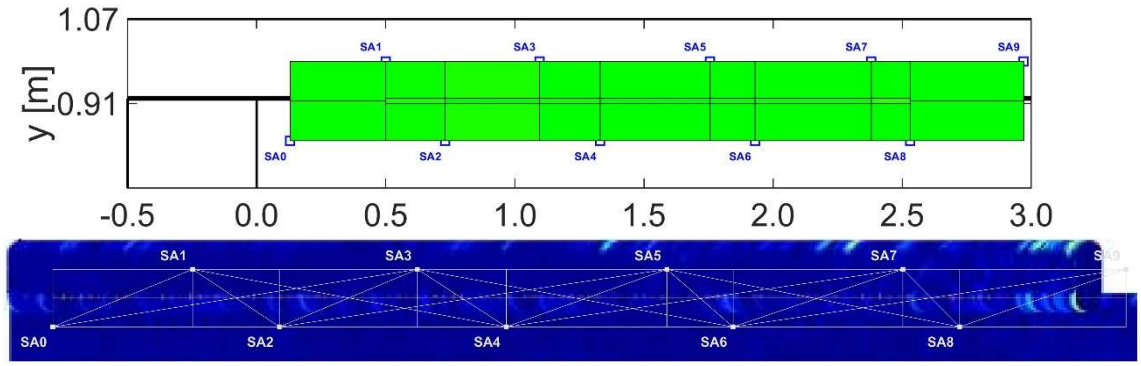
Tomography – Relative Field Velocity & Interface Time Interference (dt) vs DIC



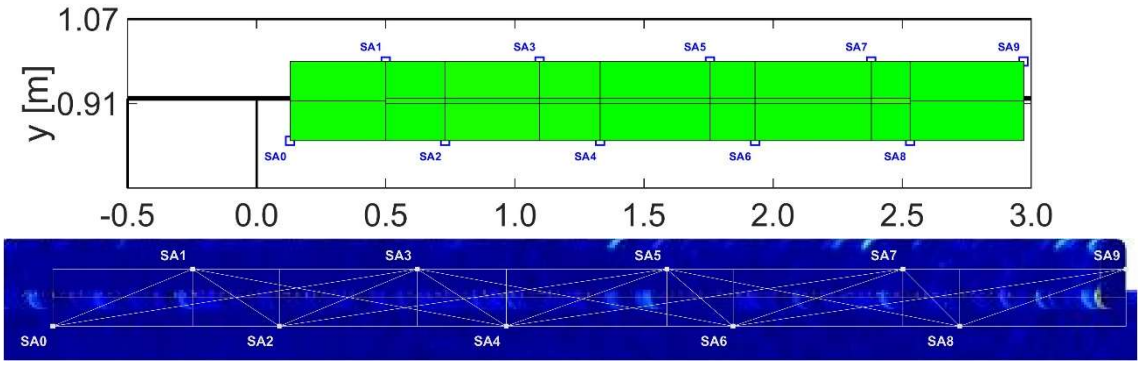
(b)-(ag)



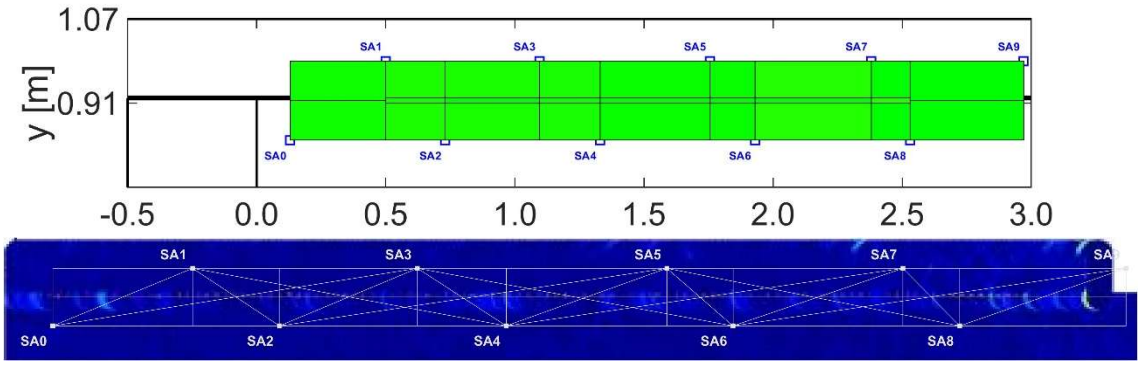
(b)- 0[kN]



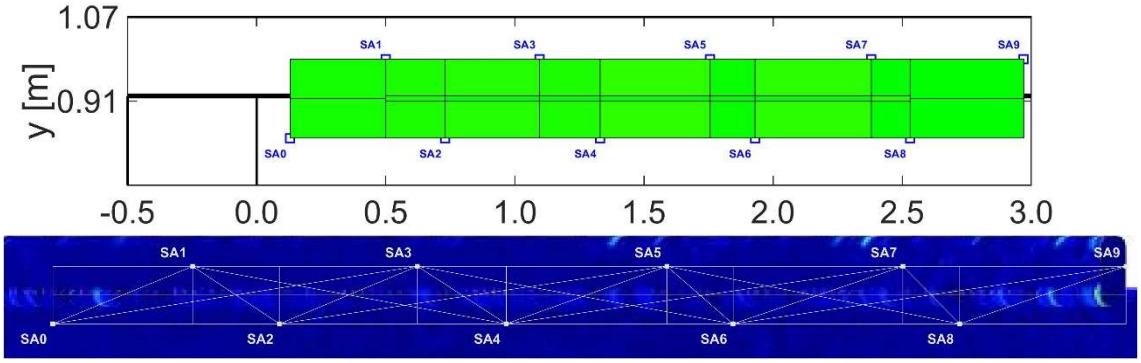
(c)- 50[kN]



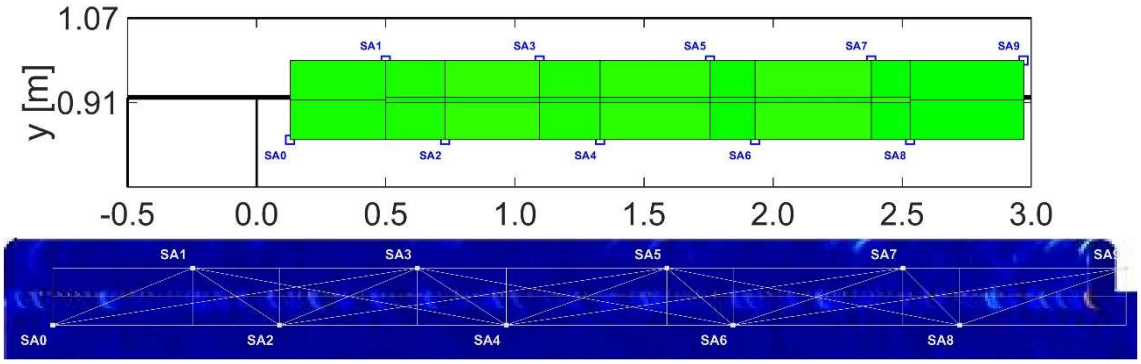
(d)- 100[kN]



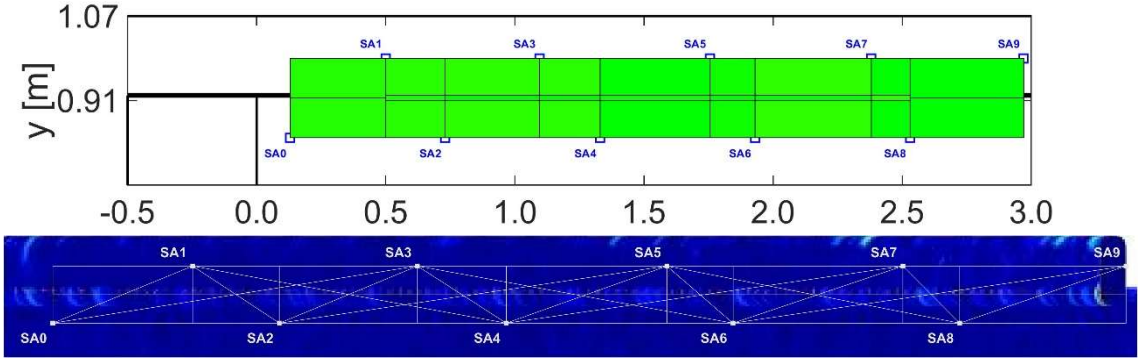
(e)- 150[kN]



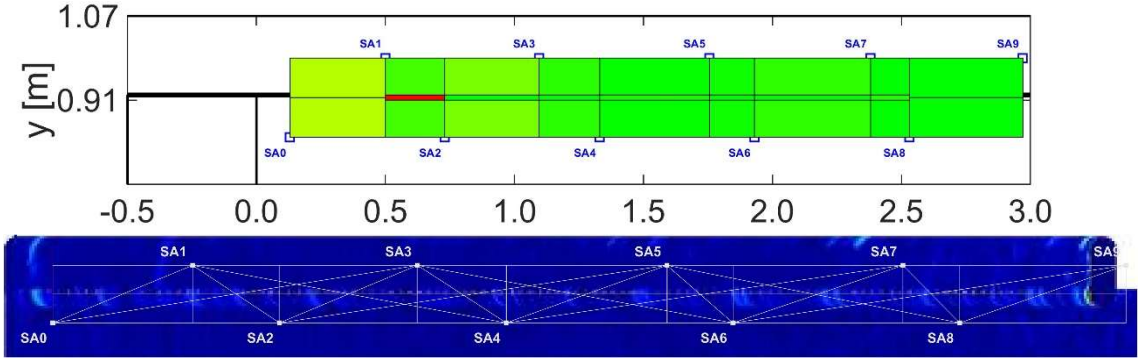
(f)- 200[kN]



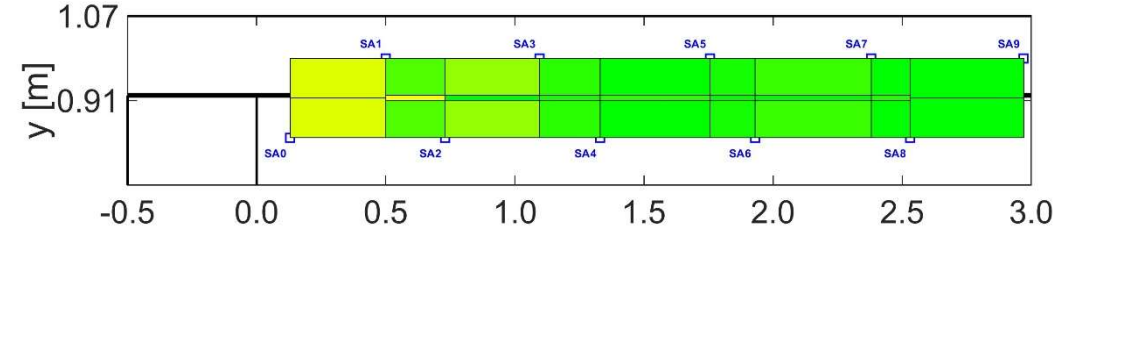
(g)- 250[kN]

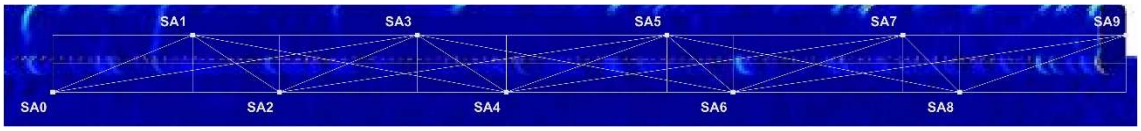


(h)- 300[kN]

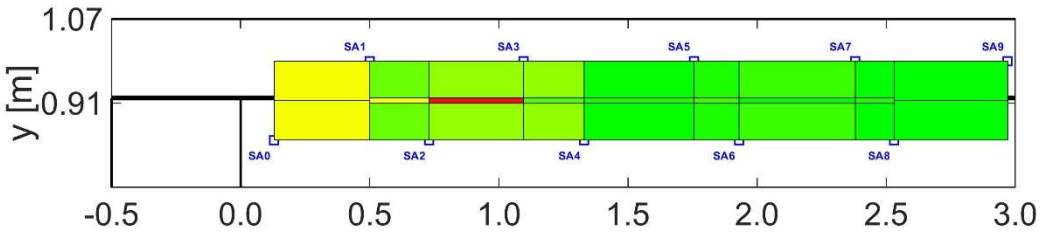


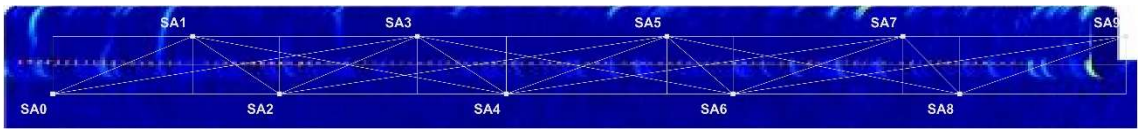
(i)- 350[kN]



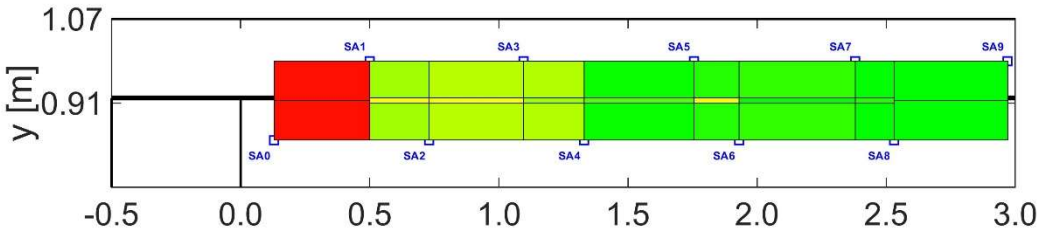


(j)- 400[kN]



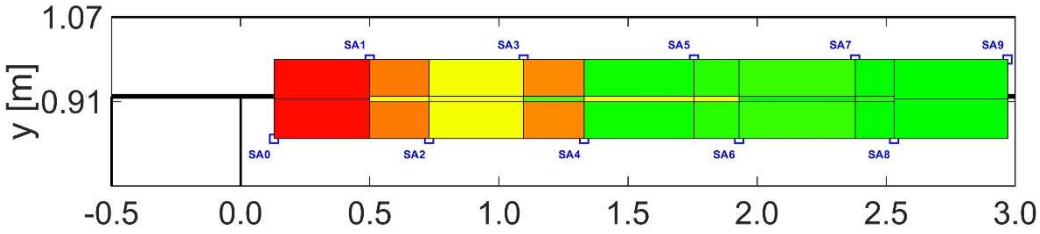


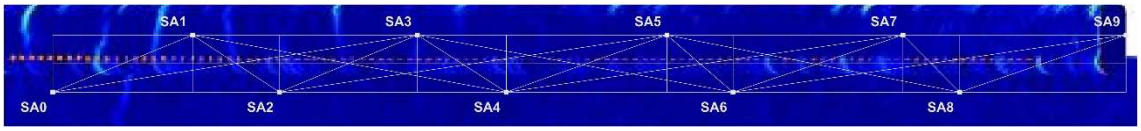
(k)- 450[kN]



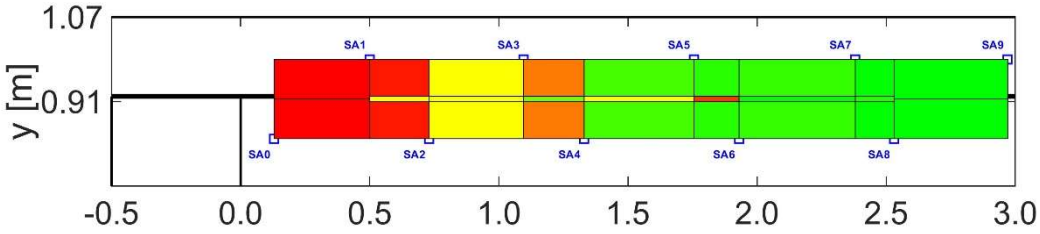
(no DIC Figure)

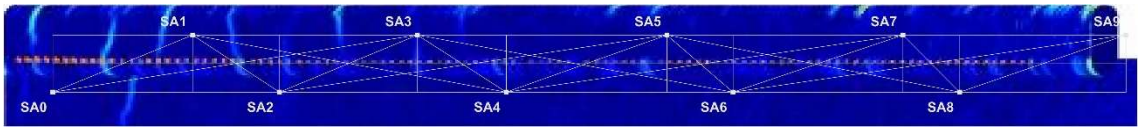
(l)- 500[kN]



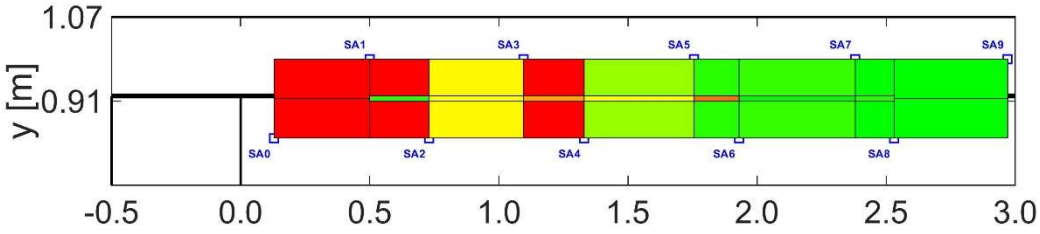


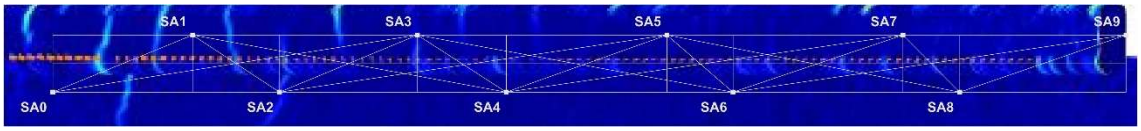
(m)- 550[kN]



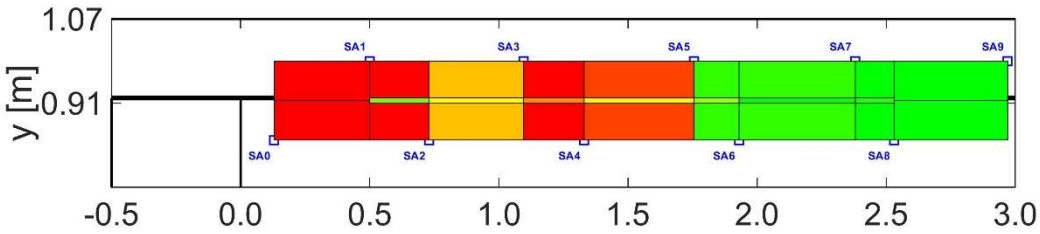


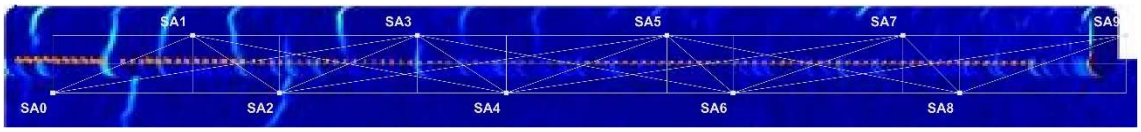
(n)- 600[kN]



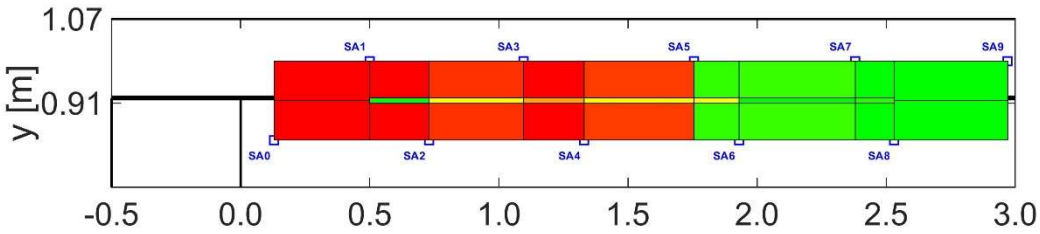


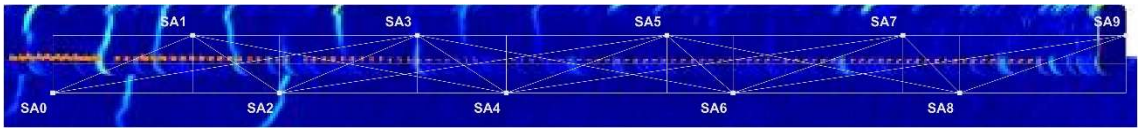
(o)- 650[kN]



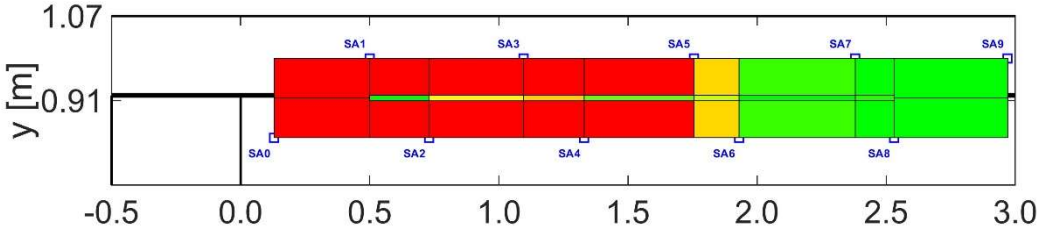


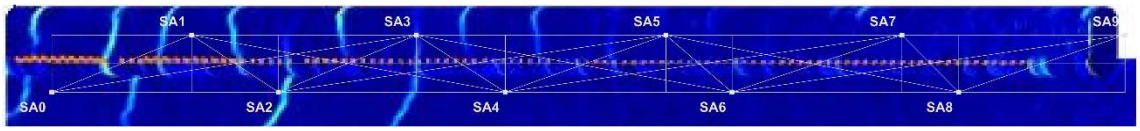
(p)- 700[kN]



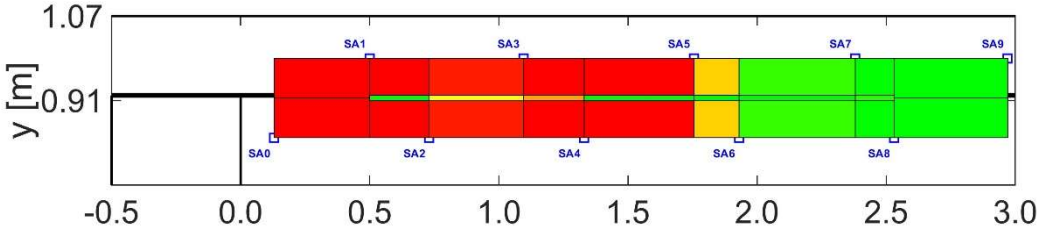


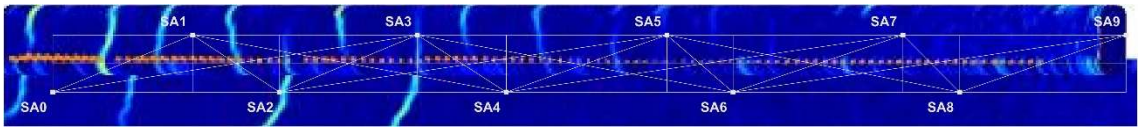
(q)- 750[kN]



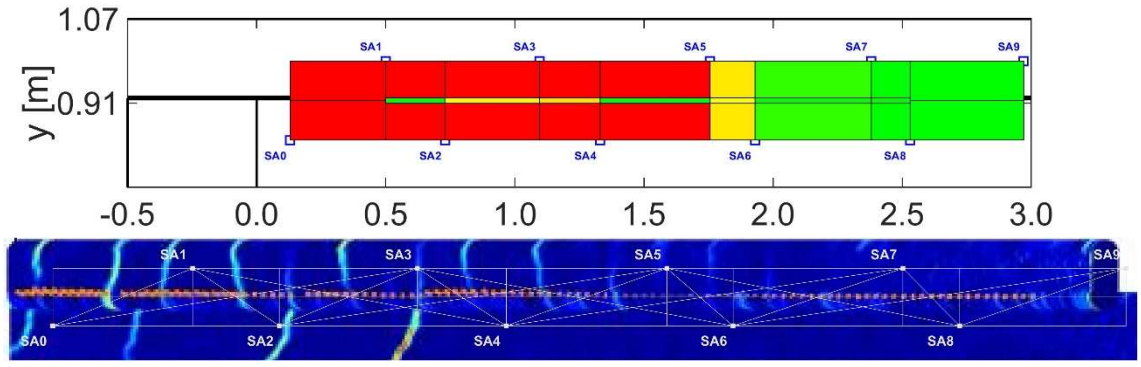


(r)- 800[kN]

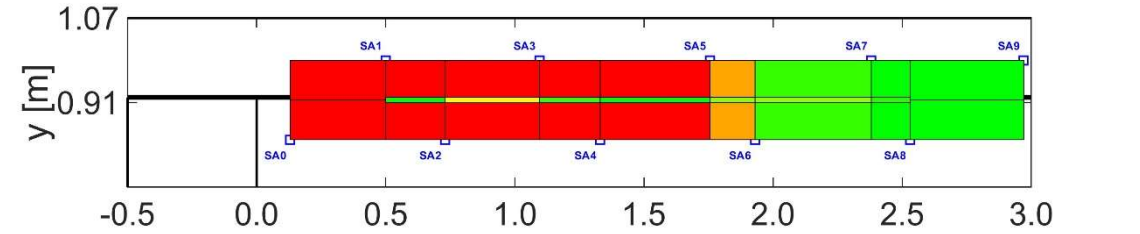




(s)- 850[kN]

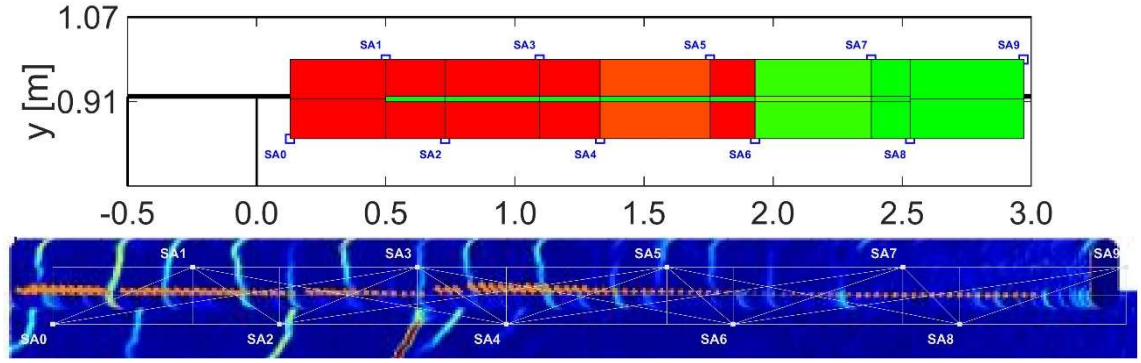


(t)- 900[kN]

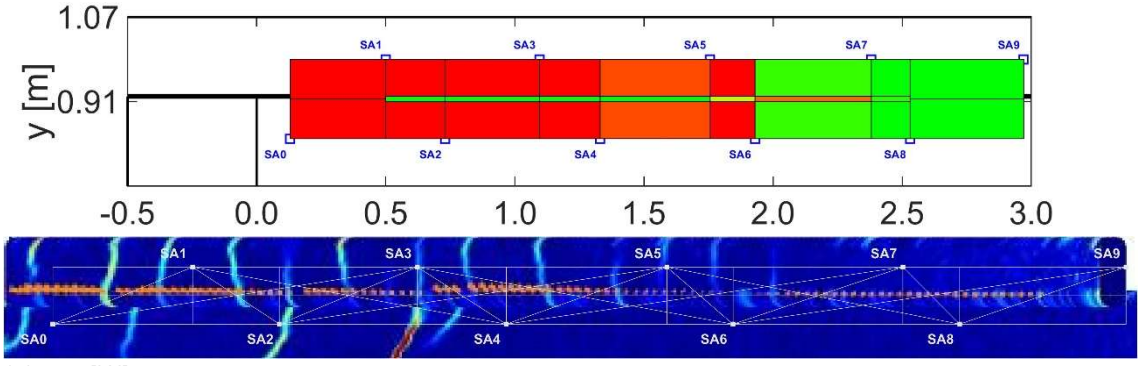


(no DIC Figure)

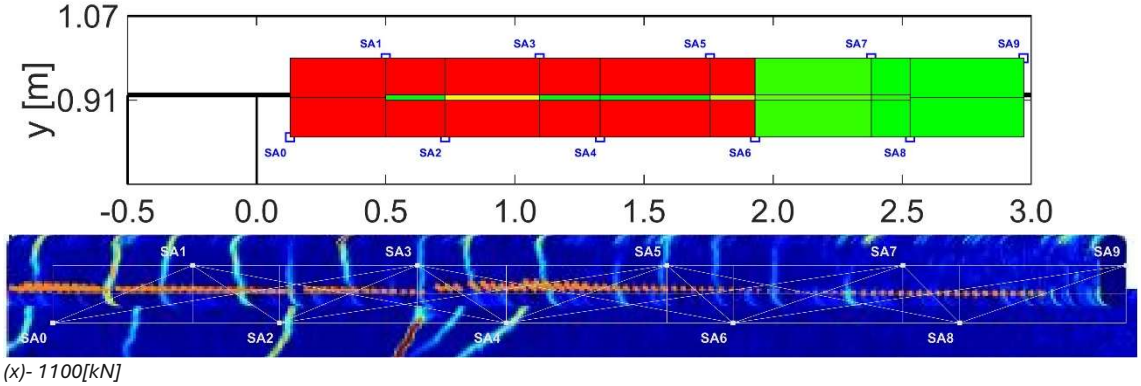
(u)- 950[kN]



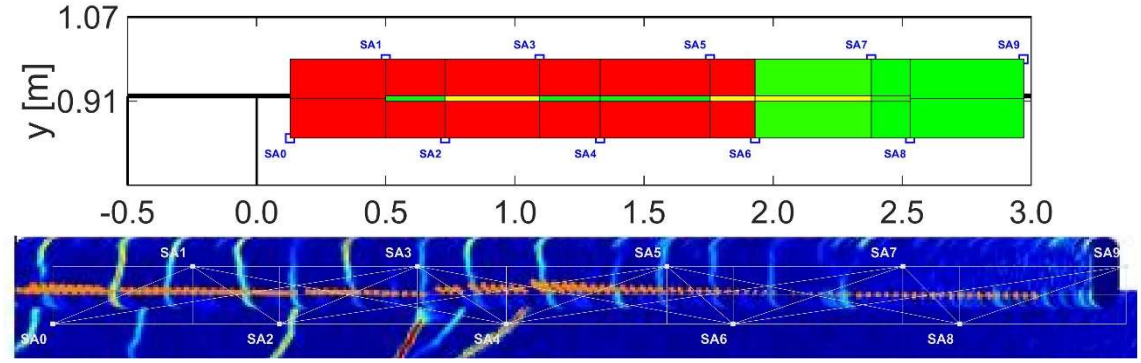
(v)- 1000[kN]



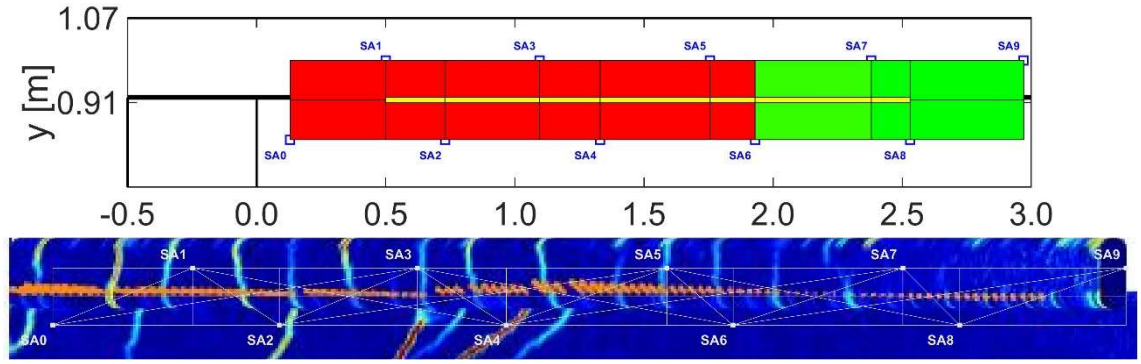
(w)- 1050[kN]



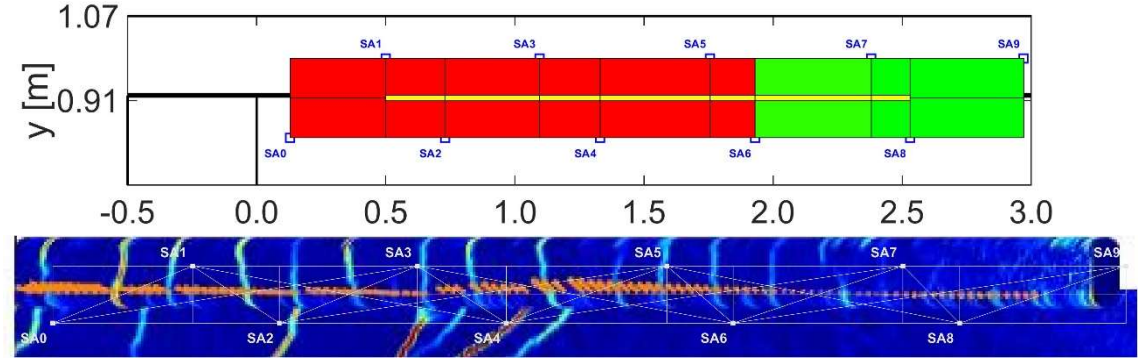
(x)- 1100[kN]



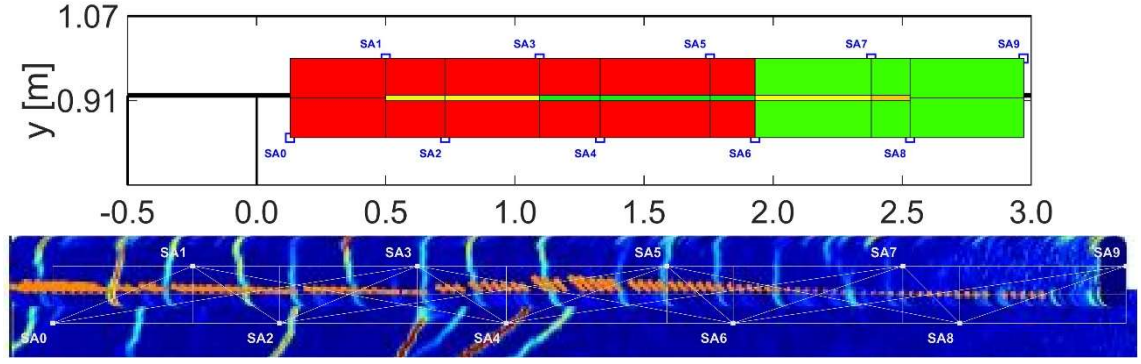
(y)- 1150[kN]



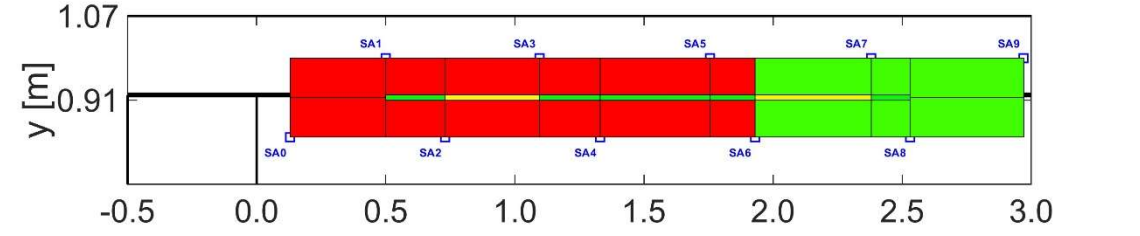
(z)- 1200[kN]

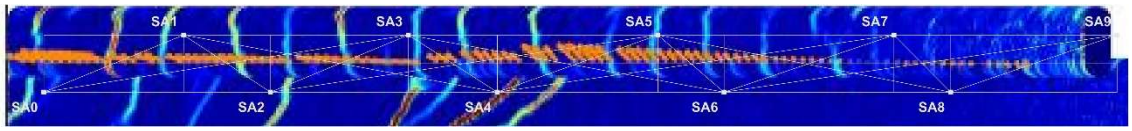


(aa)- 1250[kN]

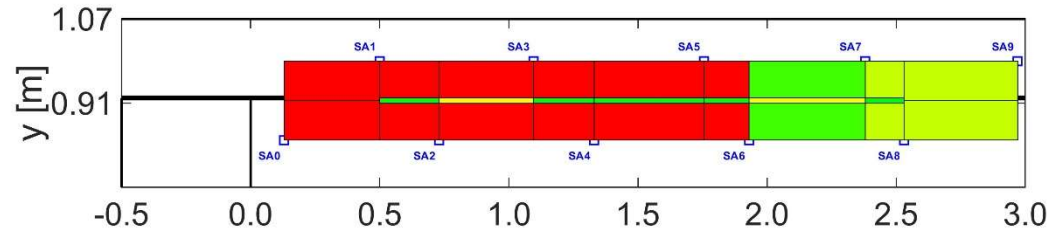


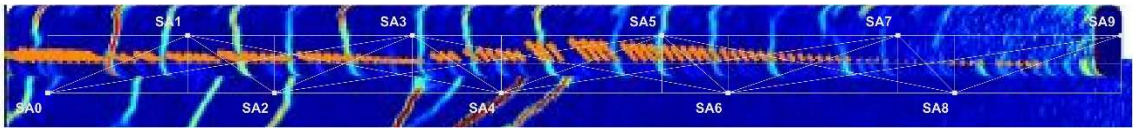
(ab)- 1300[kN]



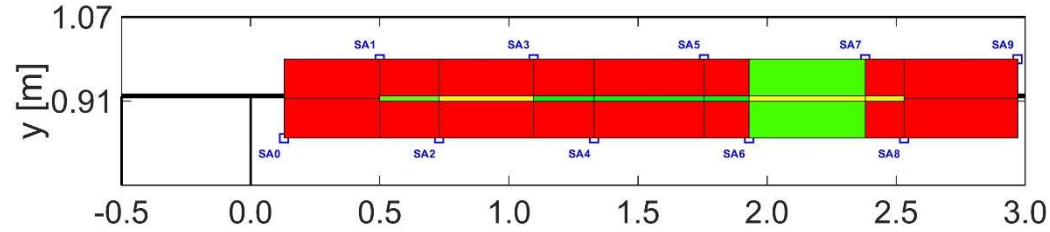


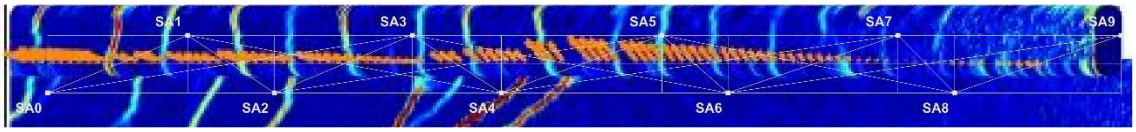
(ac)- 1350[kN]



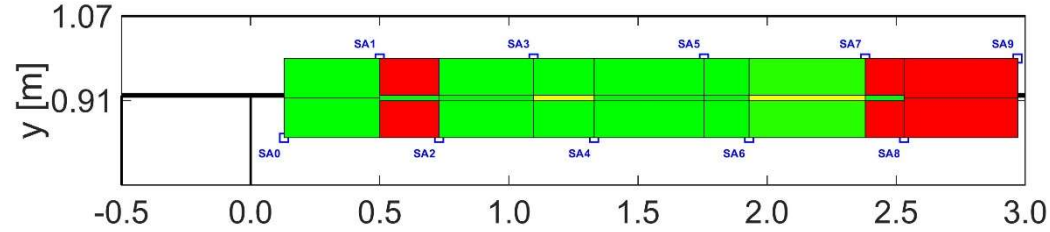


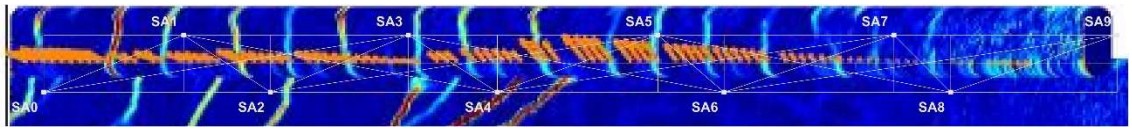
(ad)- 1400[kN]



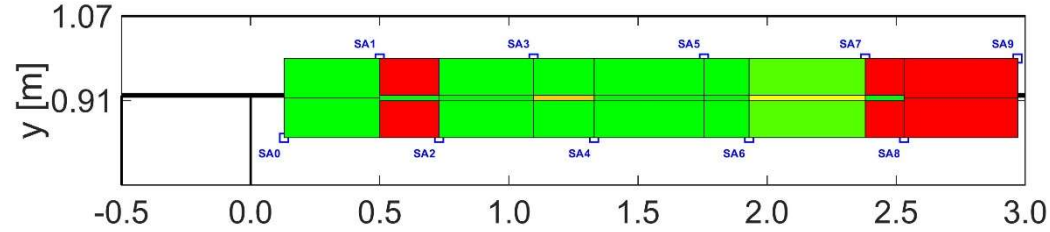


(ae)- 1450[kN]





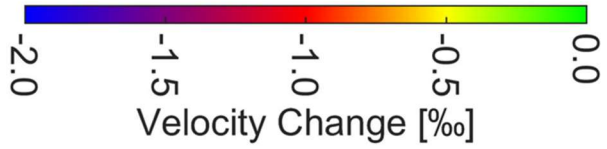
(af)- 1500[kN]



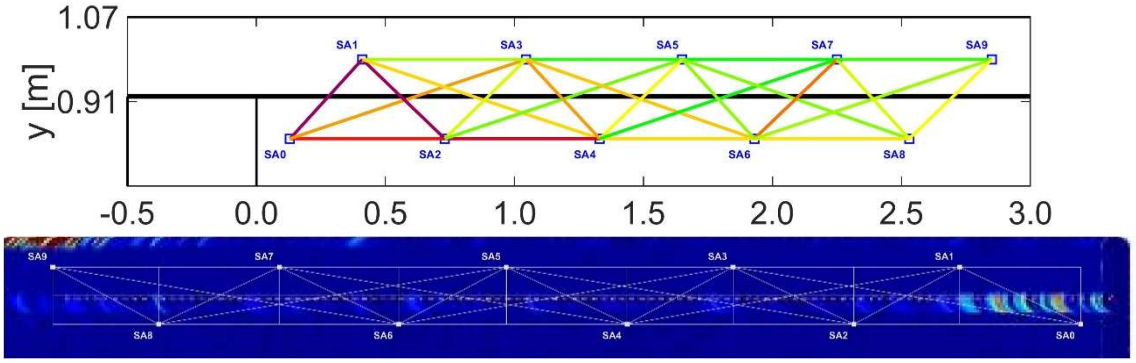
(no DIC Figure)
(ag)- 0[kN]

Appendix V.2 – beam S10H2A

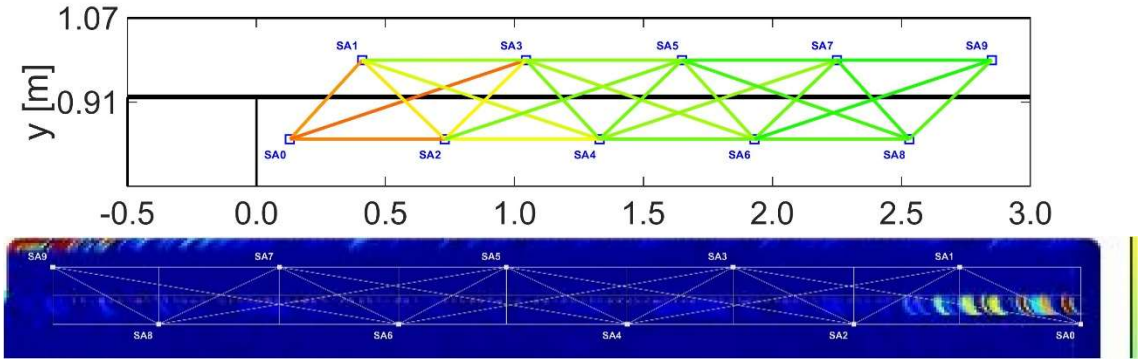
CWI – Relative Velocity Change (ε) vs DIC



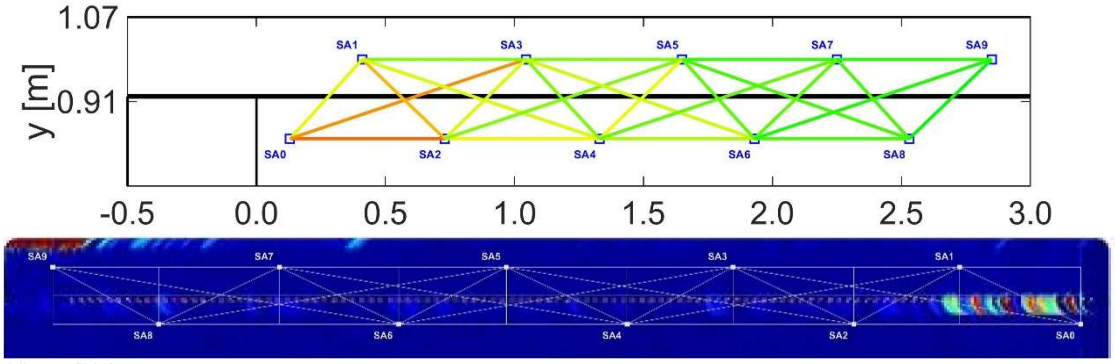
(b)-(o)



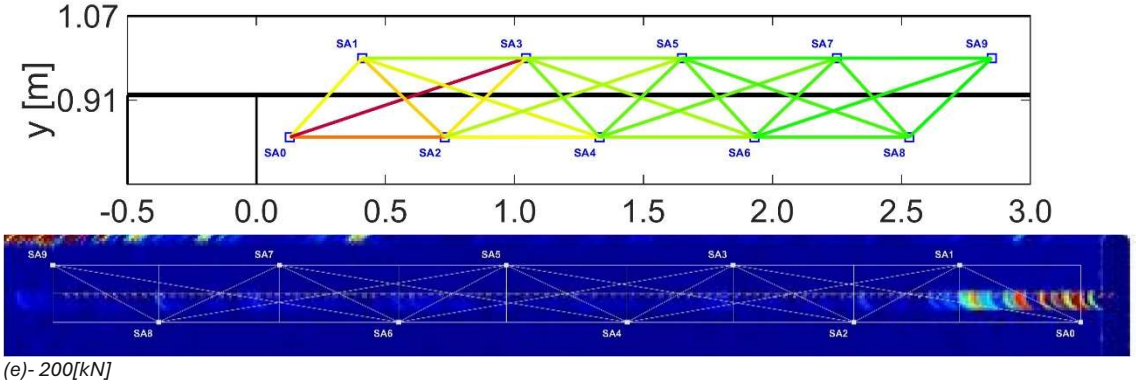
(b)- 50[kN]



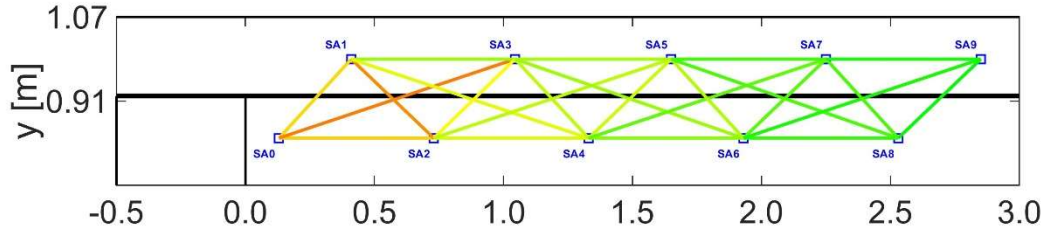
(c)- 100[kN]



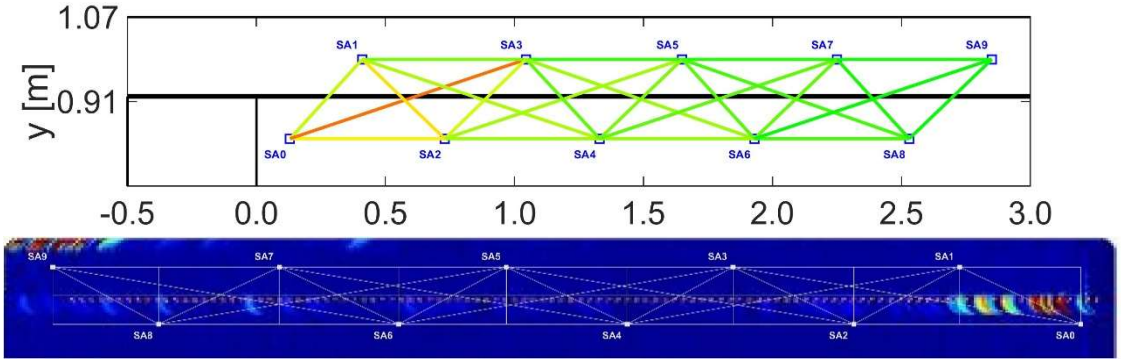
(d)- 150[kN]



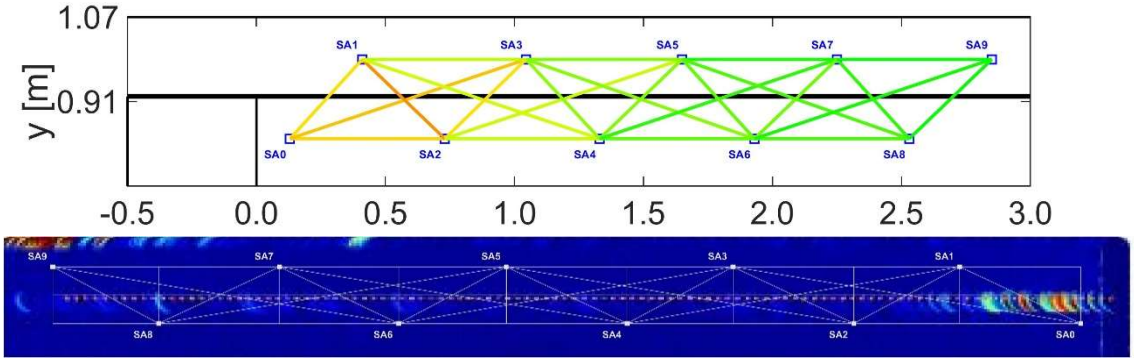
(e)- 200[kN]



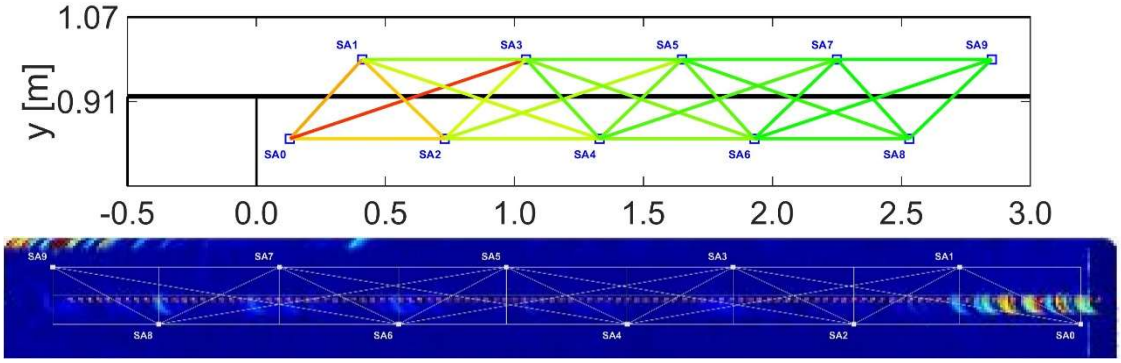
(no DIC Figure)
(f)- 250[kN]



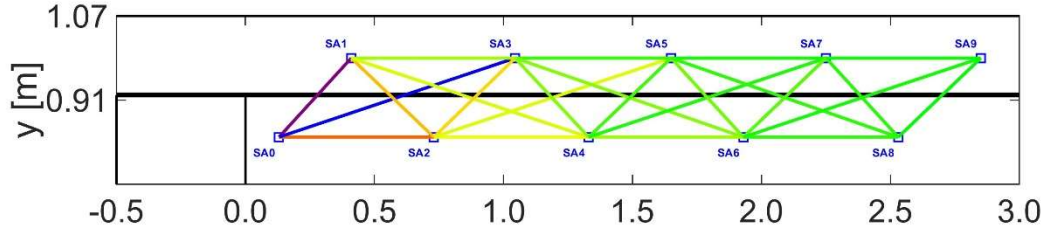
(g)- 300[kN]



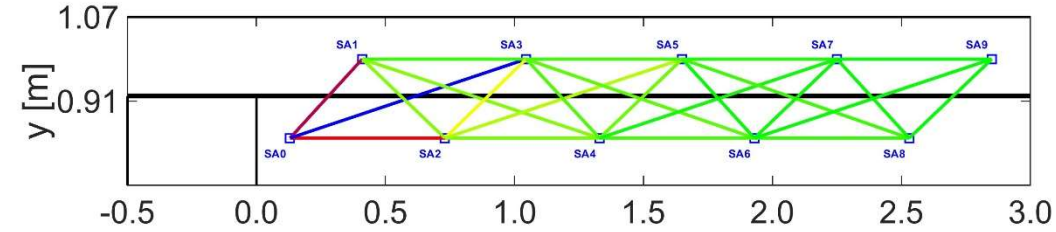
(h)- 350[kN]



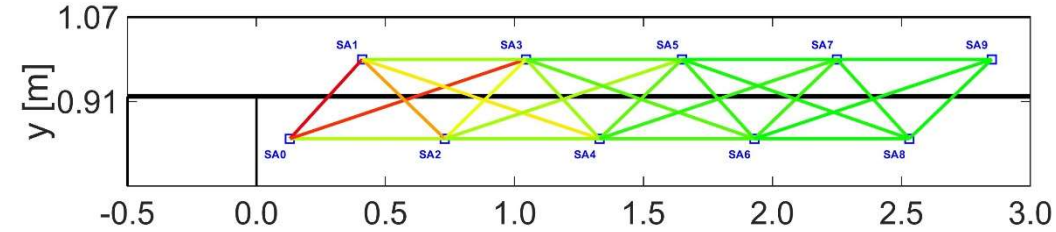
(i)- 400[kN]

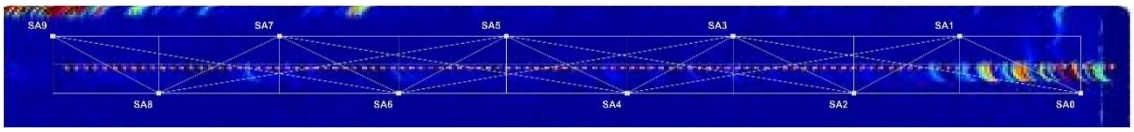


(no DIC Figure)
(j)- 450[kN]

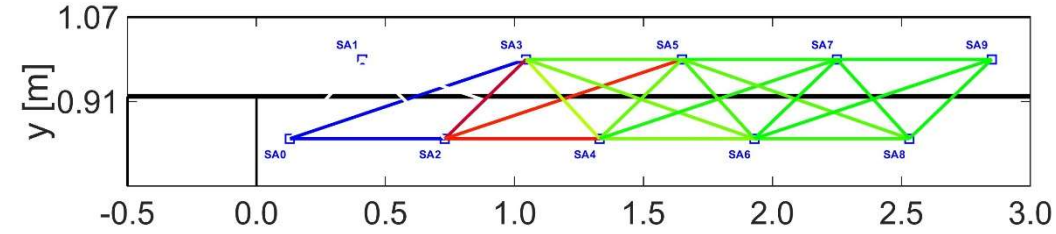


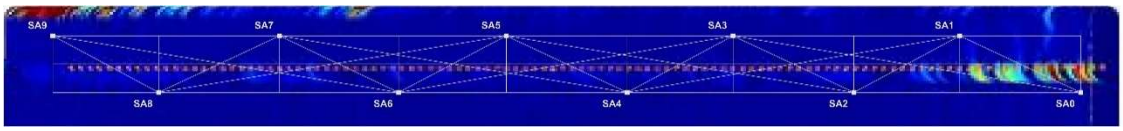
(no DIC Figure)
(k)- 500[kN]



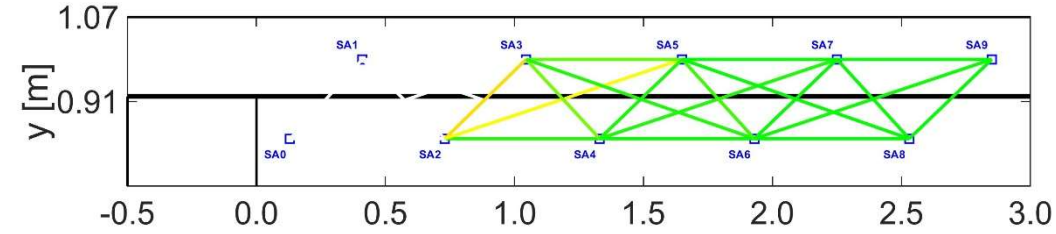


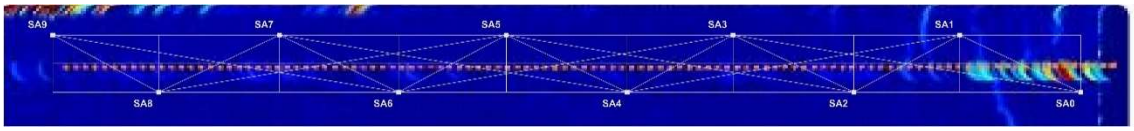
(l)- 550[kN]



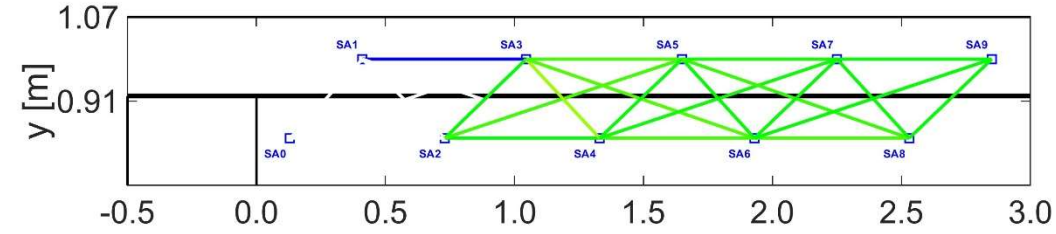


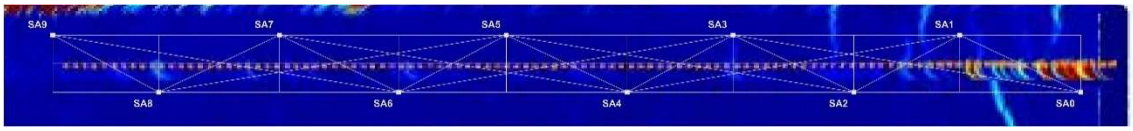
(m)- 600[kN]





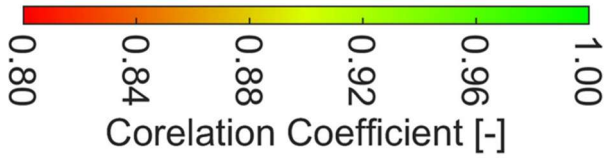
(n)- 650[kN]



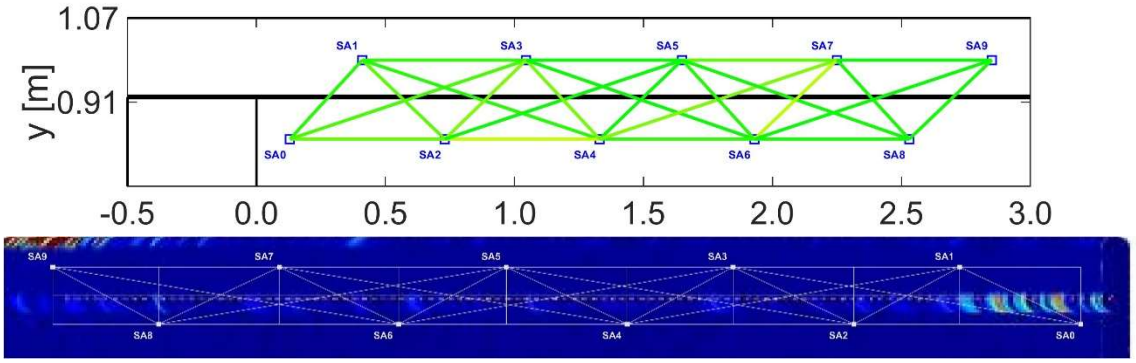


(o)- 700[kN]

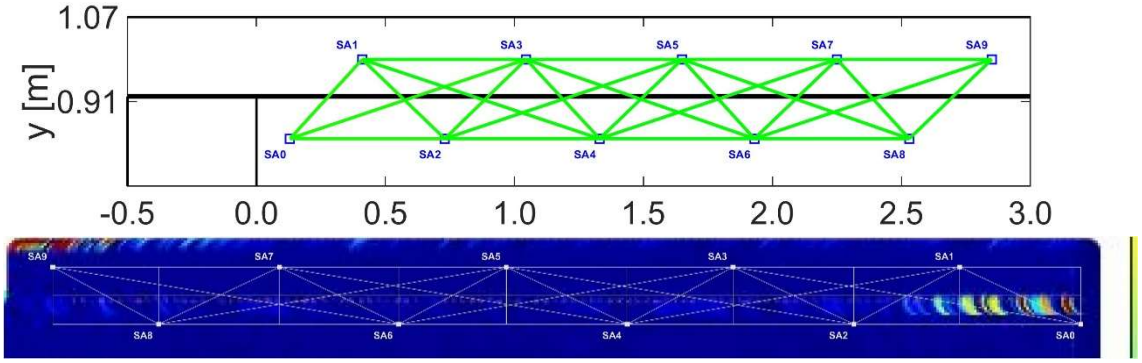
CWI – Correlation Coefficient (CC) vs DIC



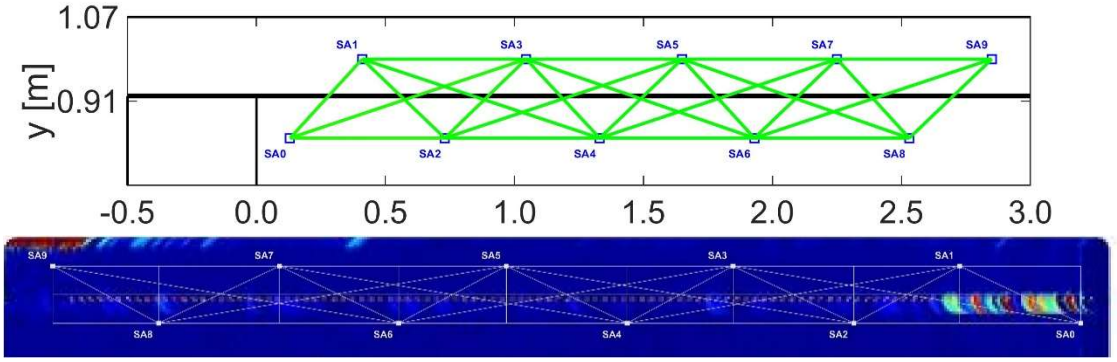
(b)-(o)



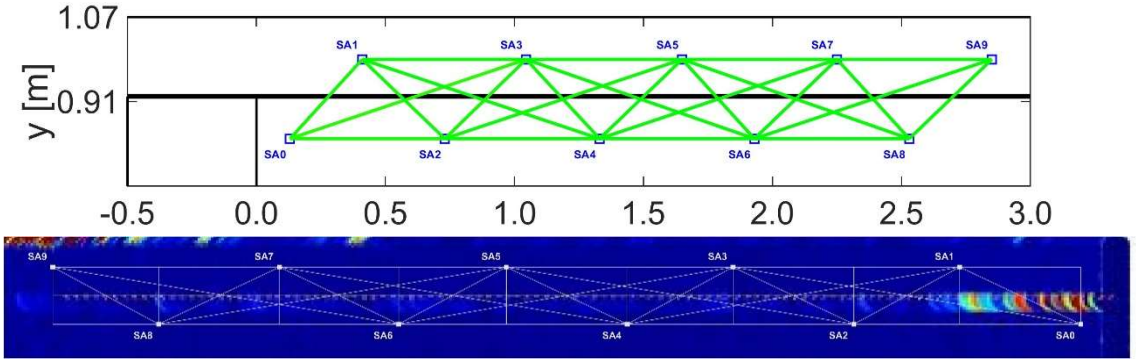
(b)- 50[kN]



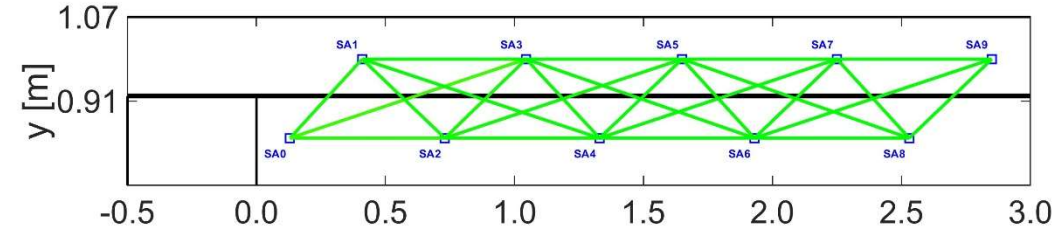
(c)- 100[kN]



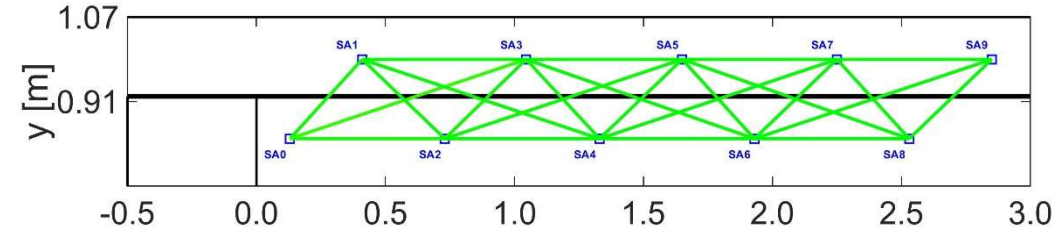
(d)- 150[kN]

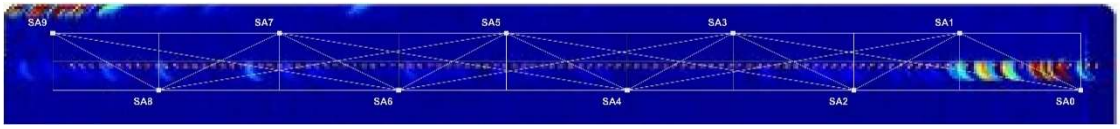


(e)- 200[kN]

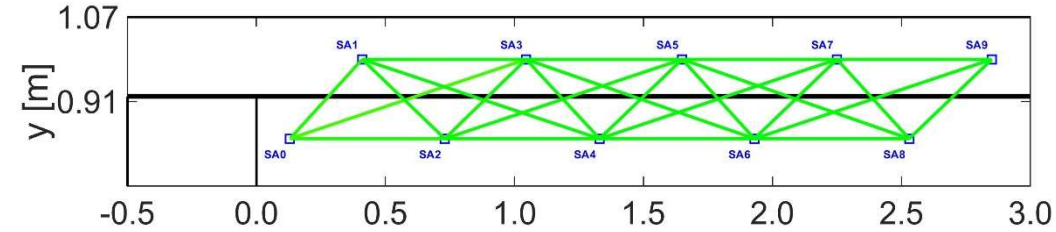


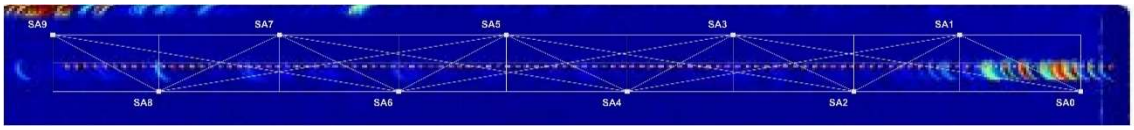
(no DIC Figure)
(f)- 250[kN]



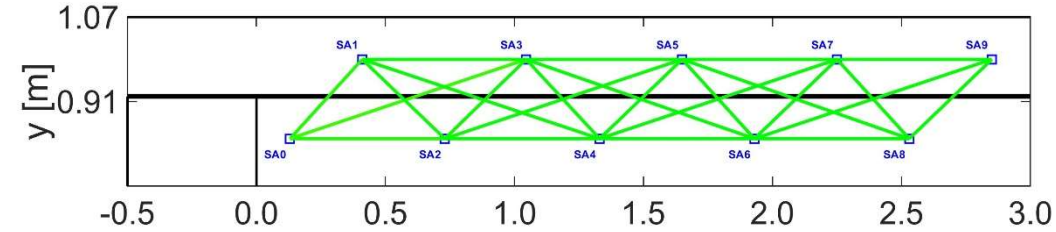


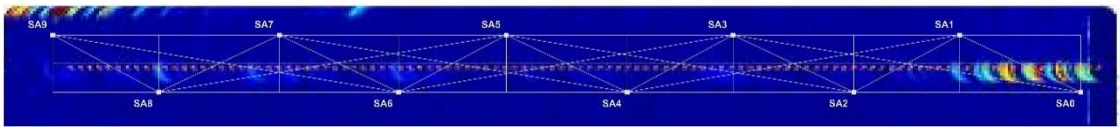
(g)- 300[kN]



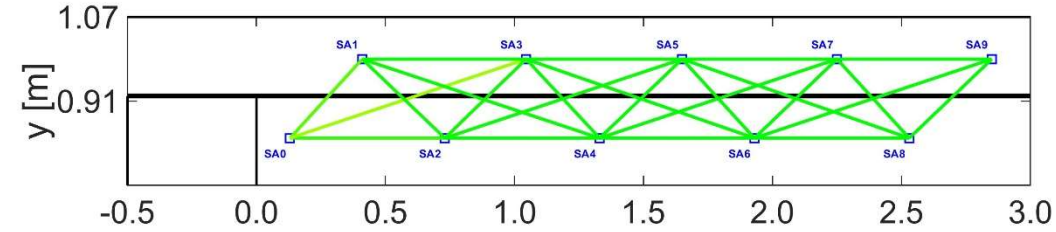


(h)- 350[kN]

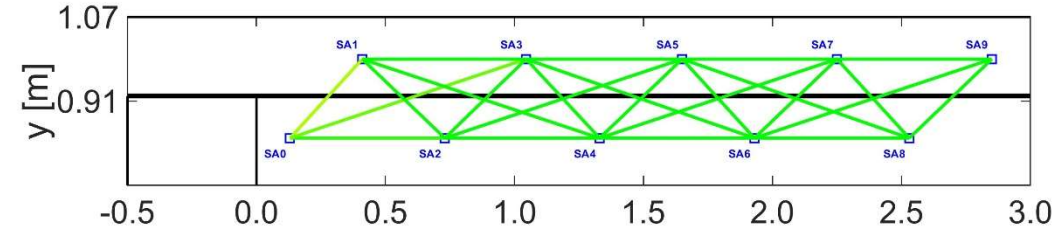




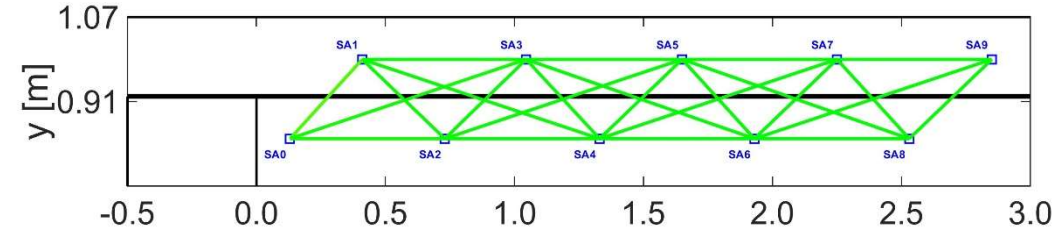
(i)- 400[kN]

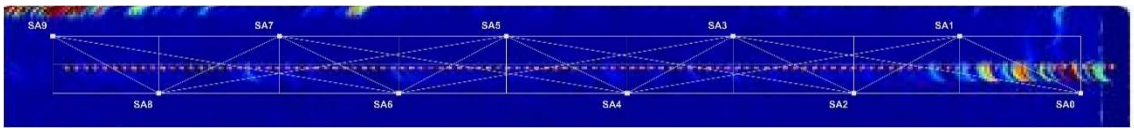


(no DIC Figure)
(j)- 450[kN]

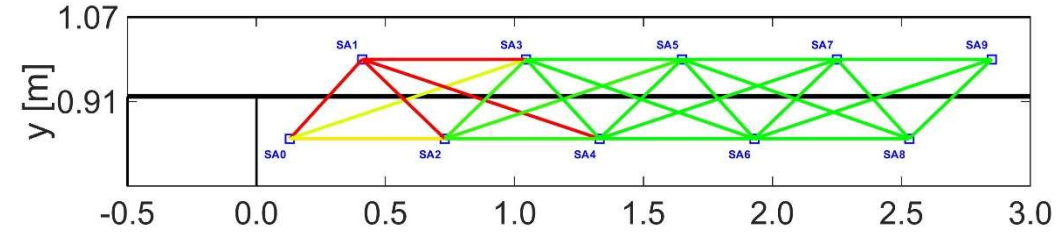


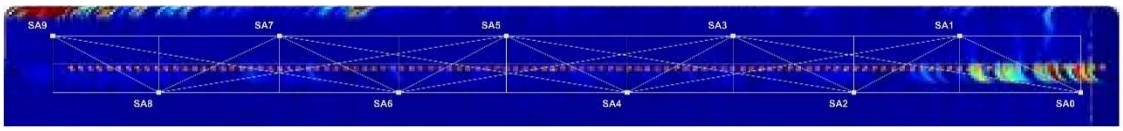
(no DIC Figure)
(k)- 500[kN]



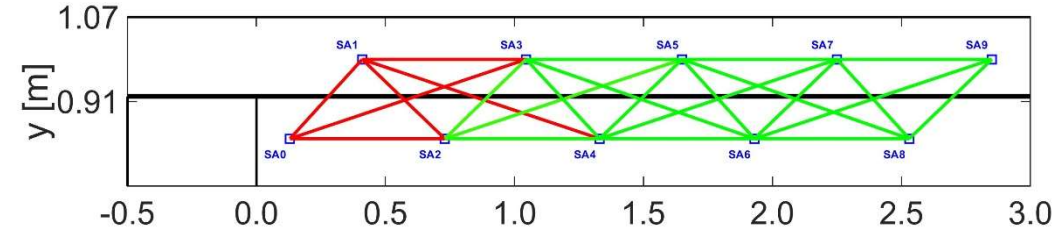


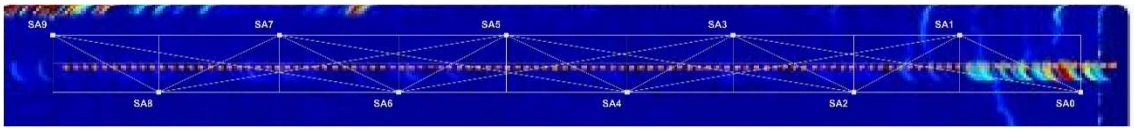
(l)- 550[kN]



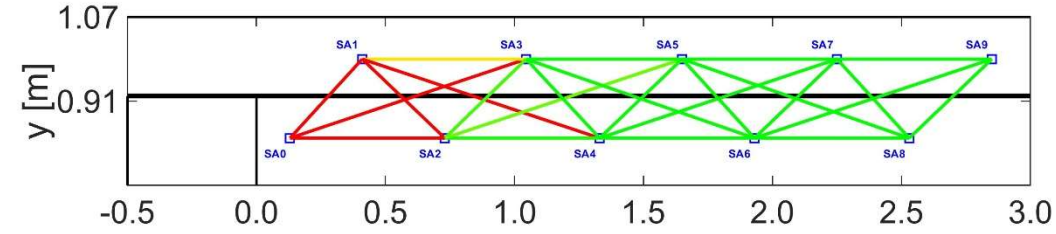


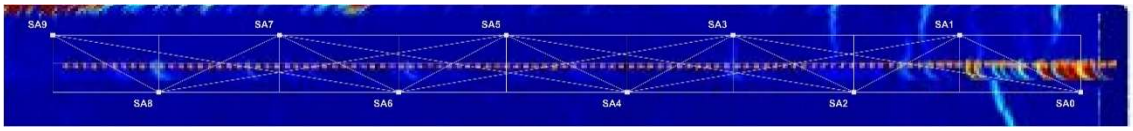
(m)- 600[kN]





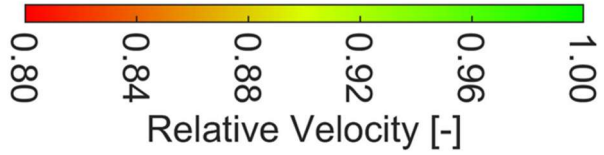
(n)- 650[kN]



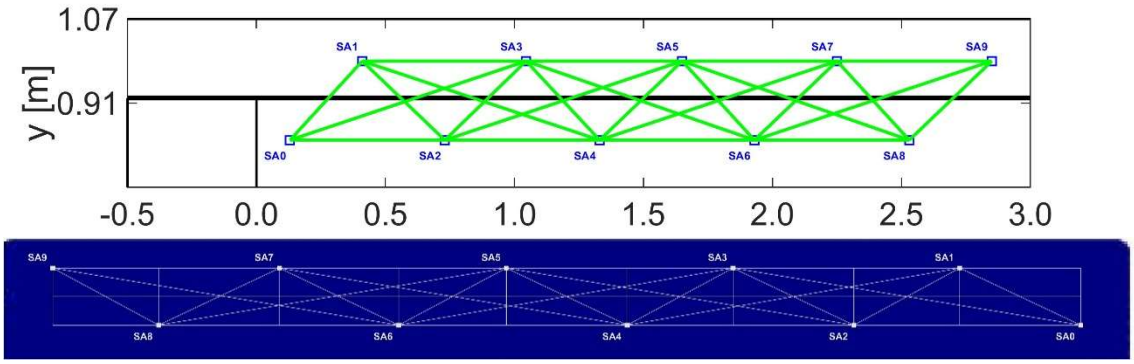


(o)- 700[kN]

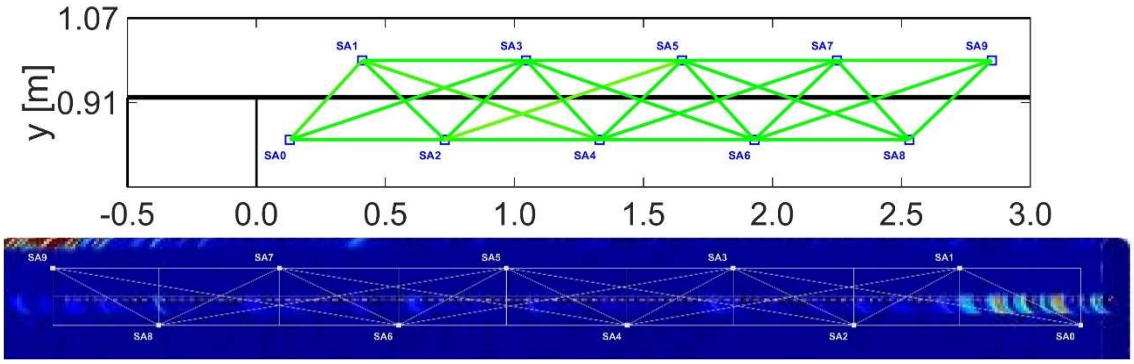
UPV – Relative Velocity vs DIC



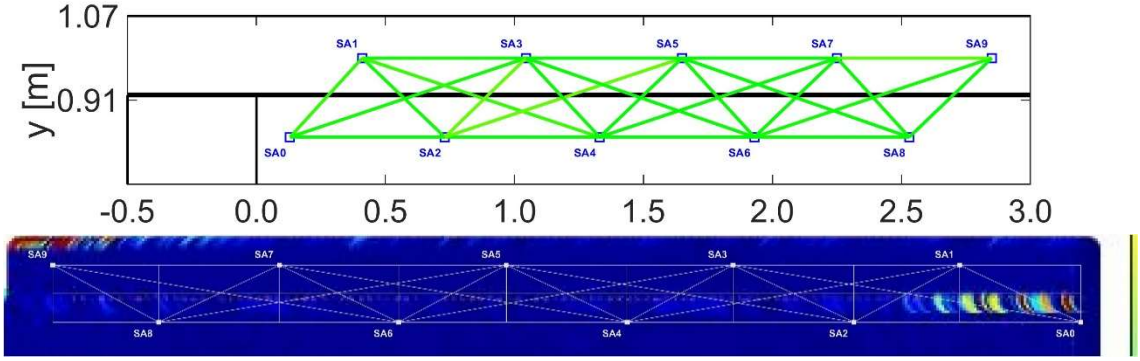
(b)-(u)



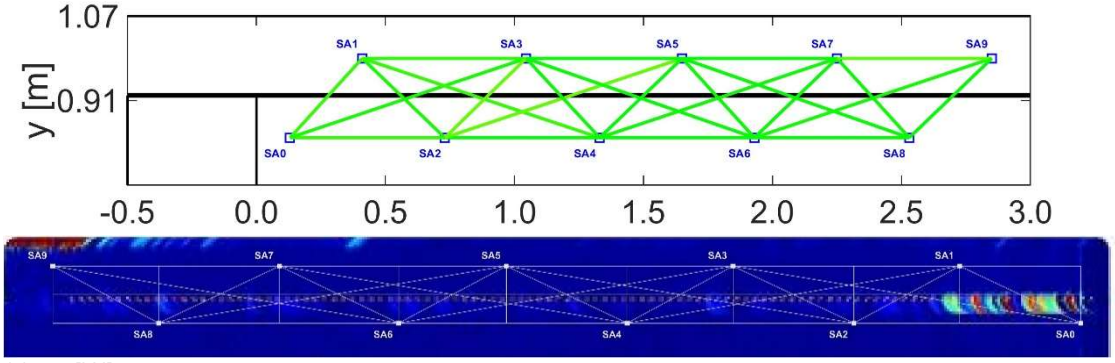
(b)- 0[kN]



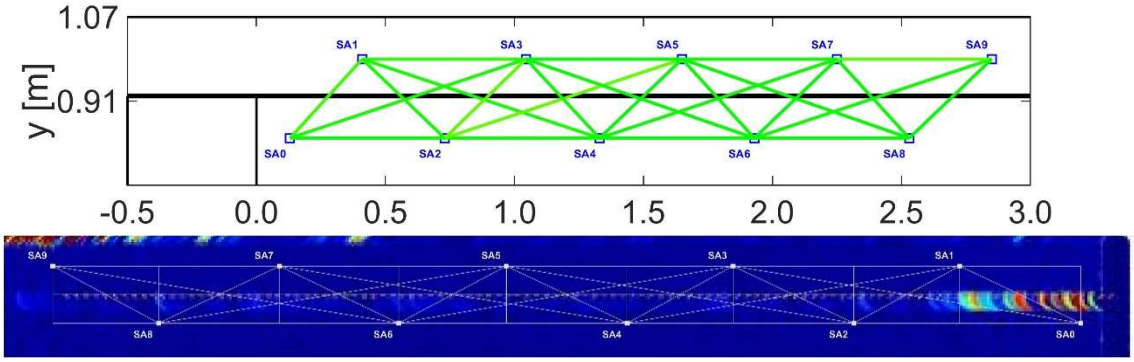
(c)- 50[kN]



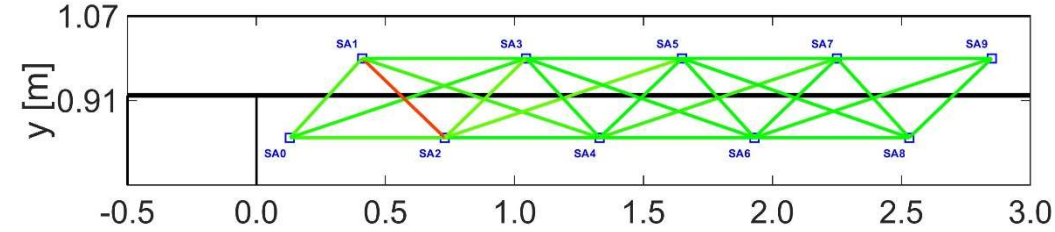
(d)- 100[kN]



(e)- 150[kN]

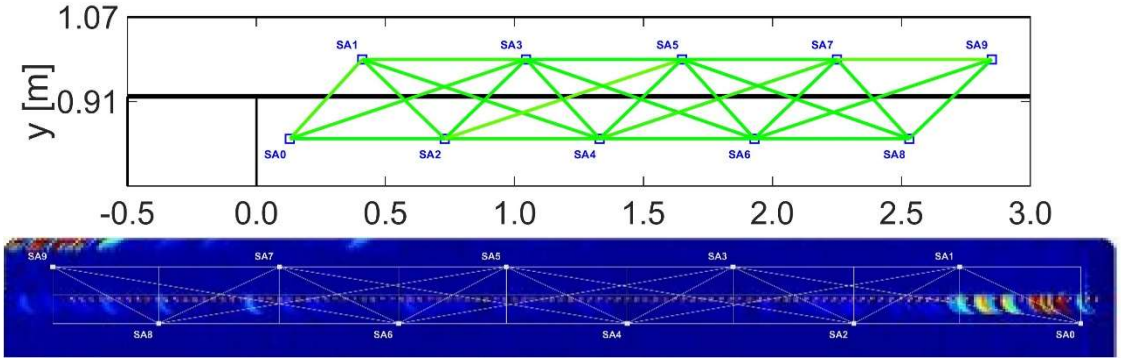


(f)- 200[kN]

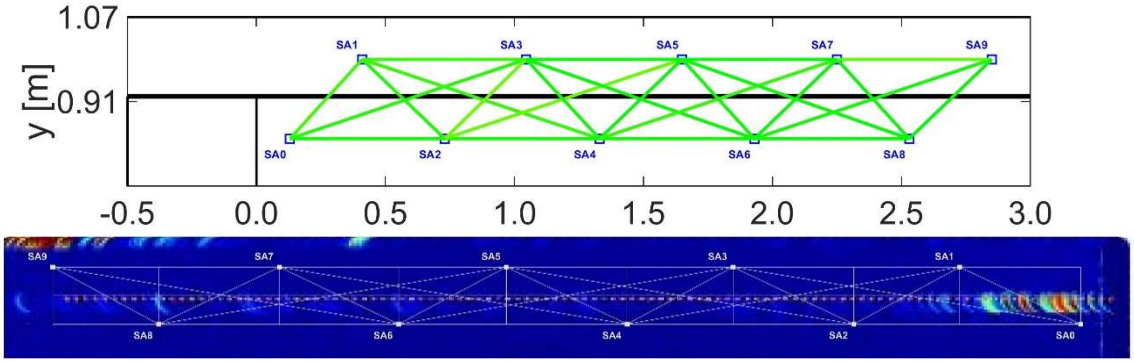


(no DIC Figure)

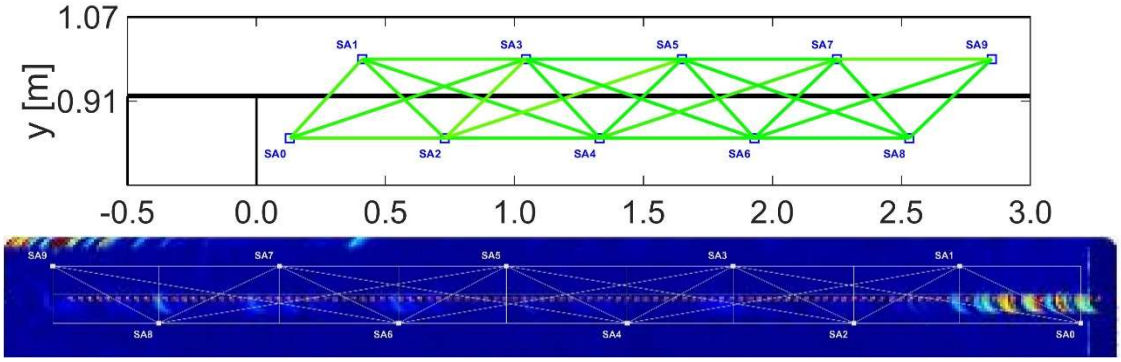
(g)- 250[kN]



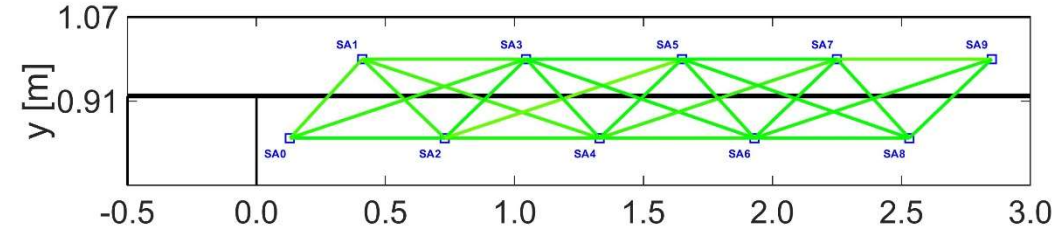
(h)- 300[kN]



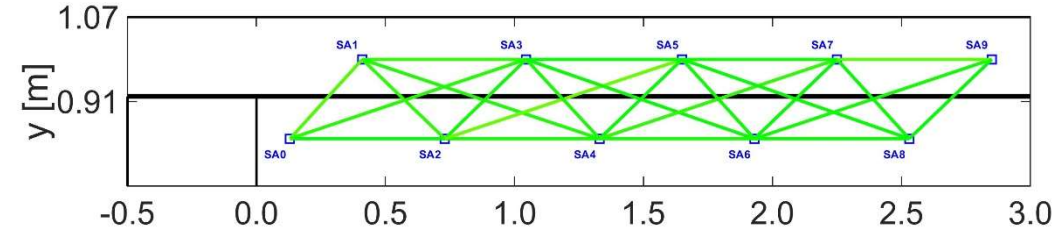
(i)- 350[kN]



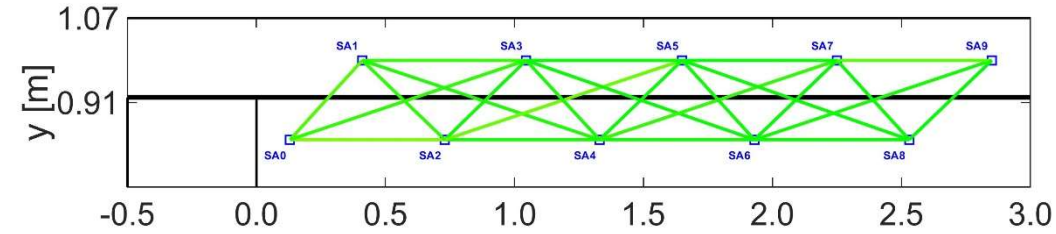
(j)- 400[kN]

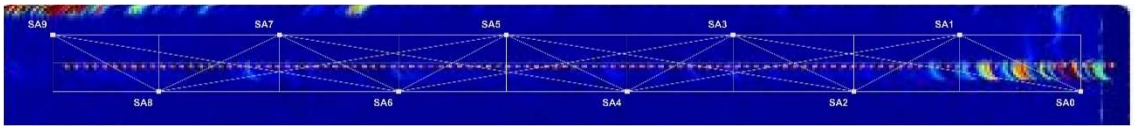


(no DIC Figure)
(k)- 450[kN]

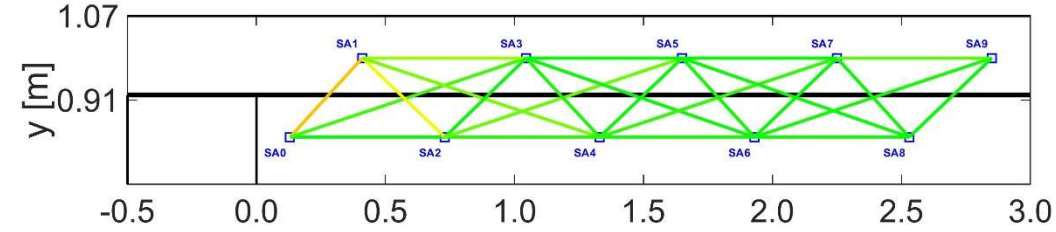


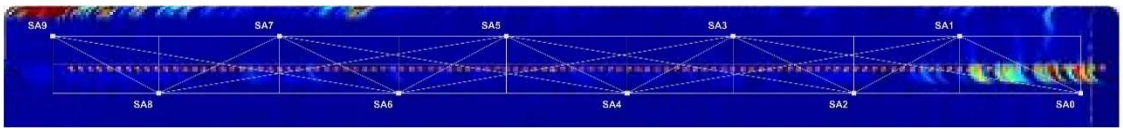
(no DIC Figure)
(l)- 500[kN]



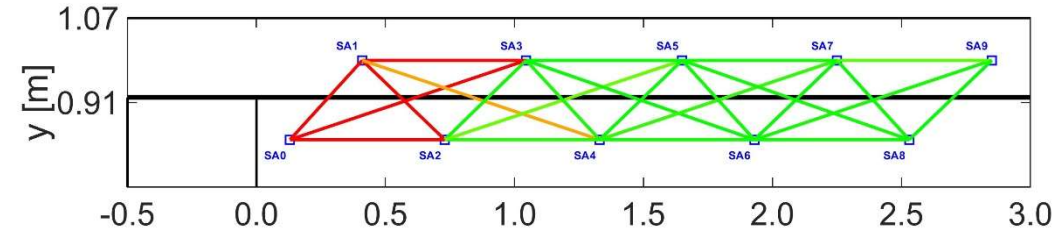


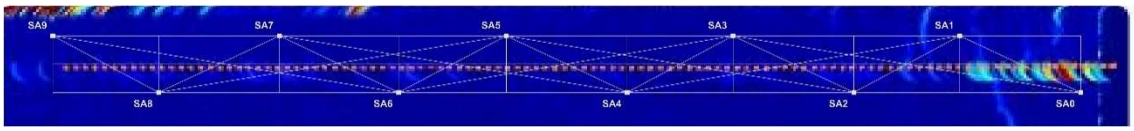
(m)- 550[kN]



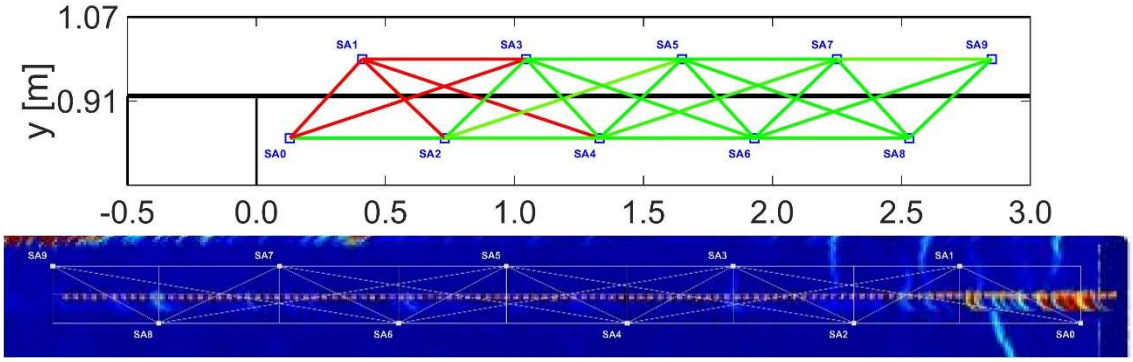


(n)- 600[kN]

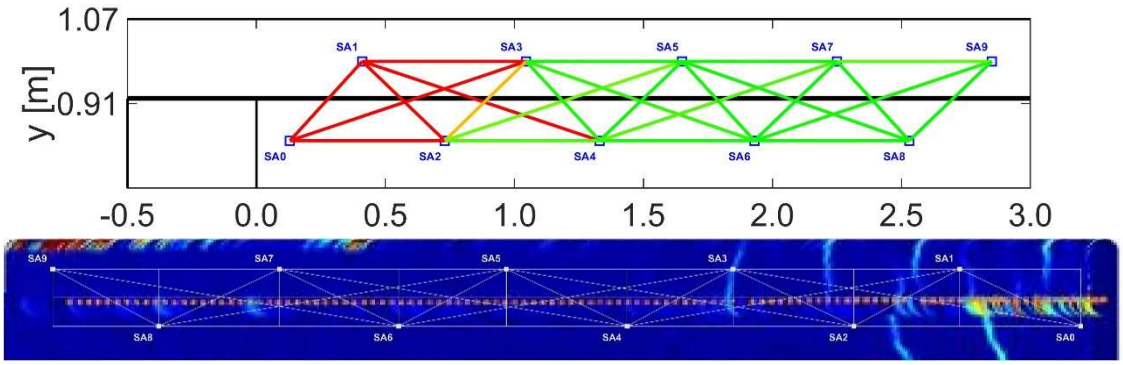




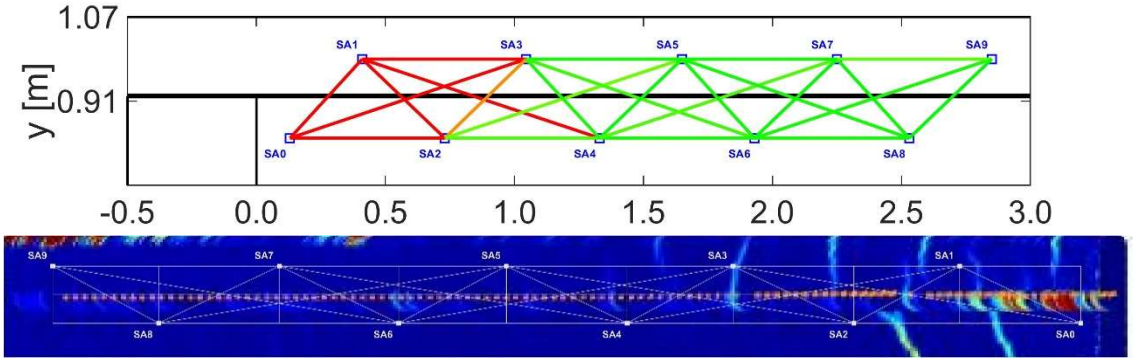
(o)- 650[kN]



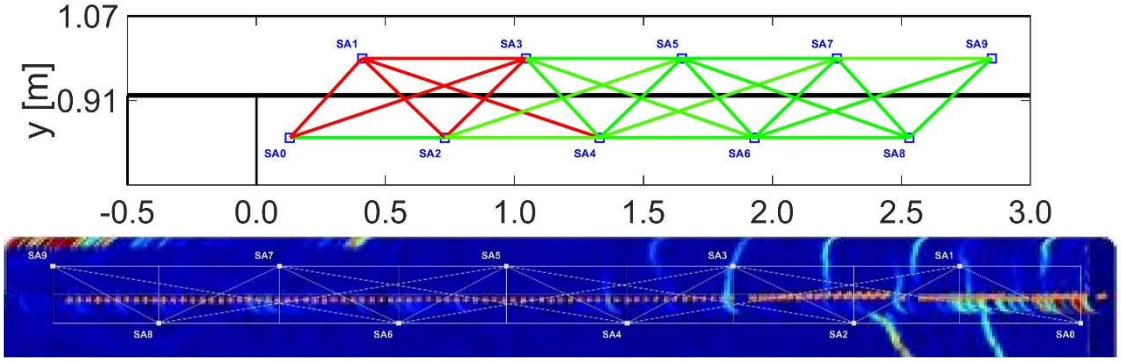
(p)- 700[kN]



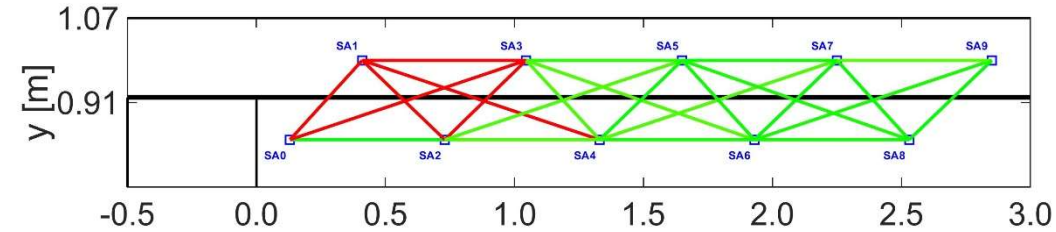
(q)- 750[kN]

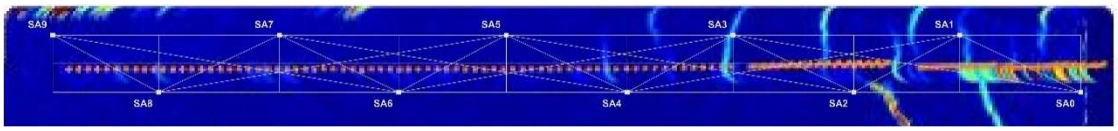


(r)- 800[kN]

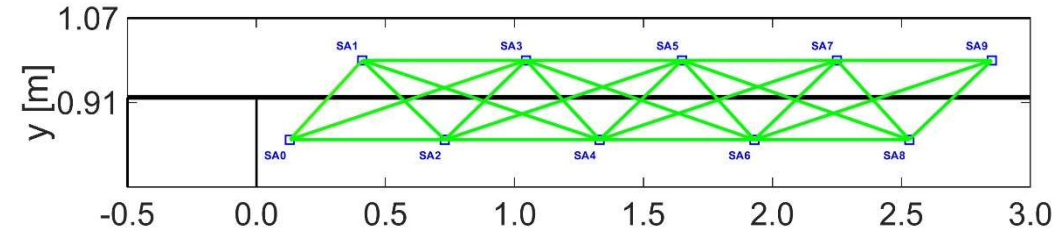


(s)- 850[kN]





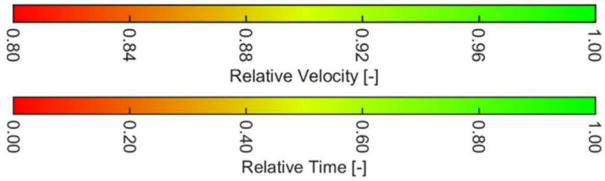
(t)- 900[kN]



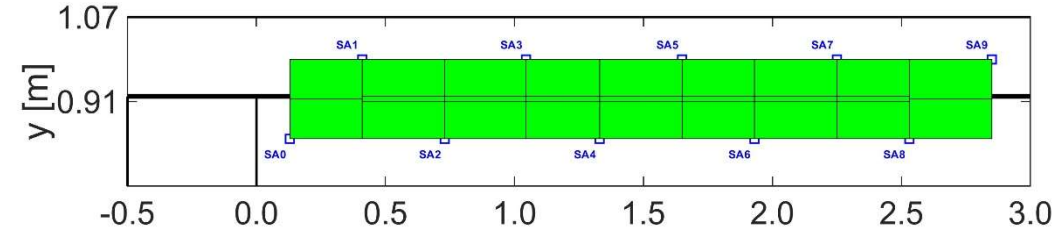
(no DIC Figure)

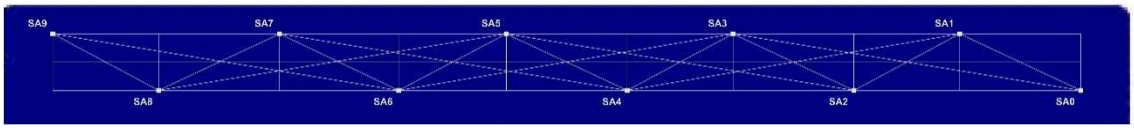
(u)- 0[kN]

Tomography – Relative Field Velocity & Interface Time Interference (dt) vs DIC

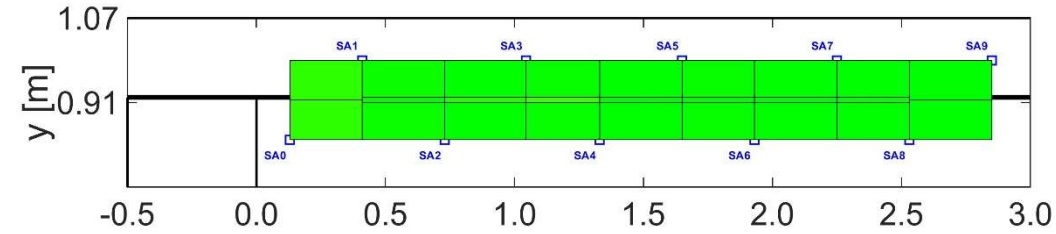


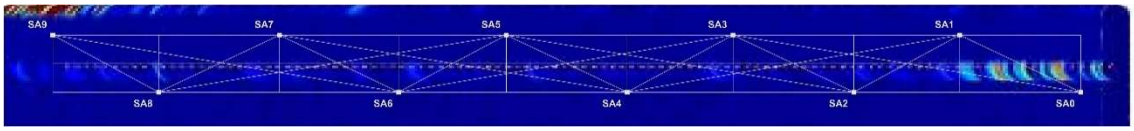
(b)-(u)



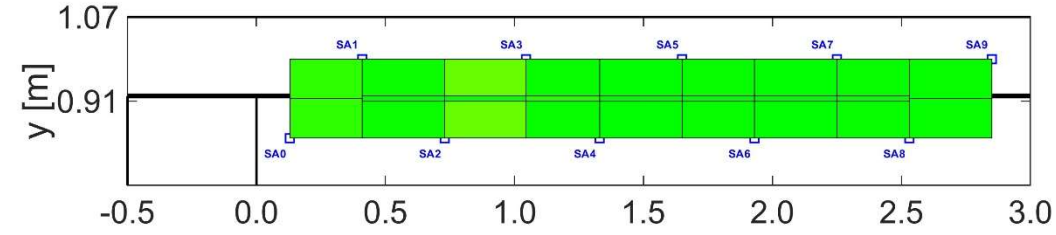


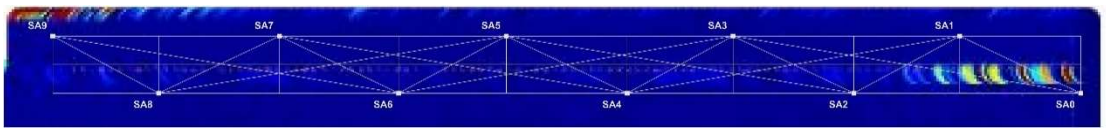
(b)- 0[kN]



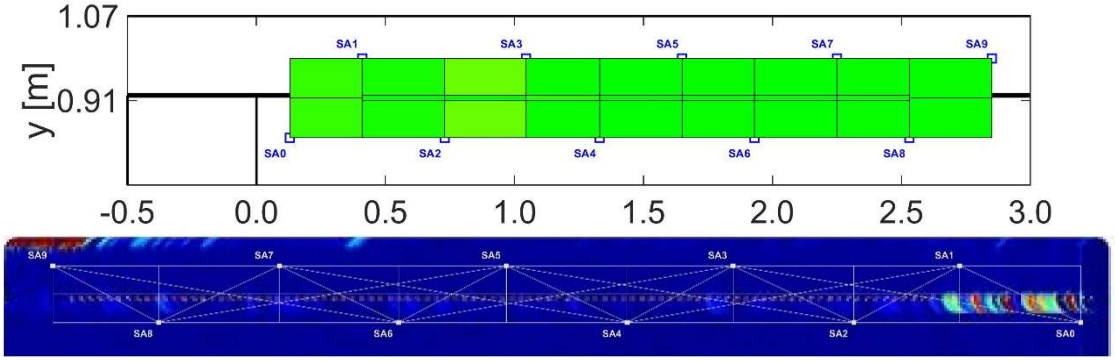


(c)- 50[kN]

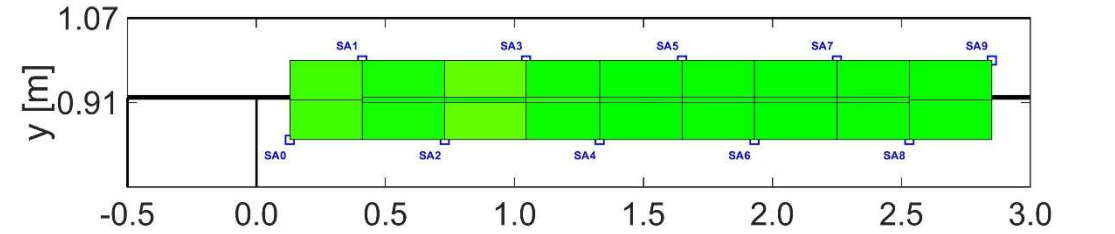


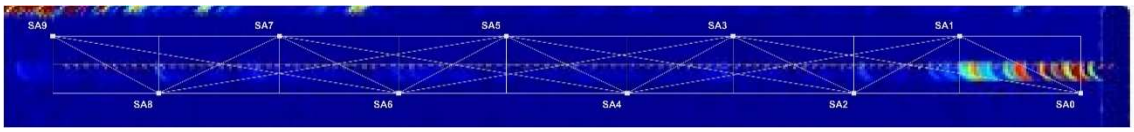


(d)- 100[kN]

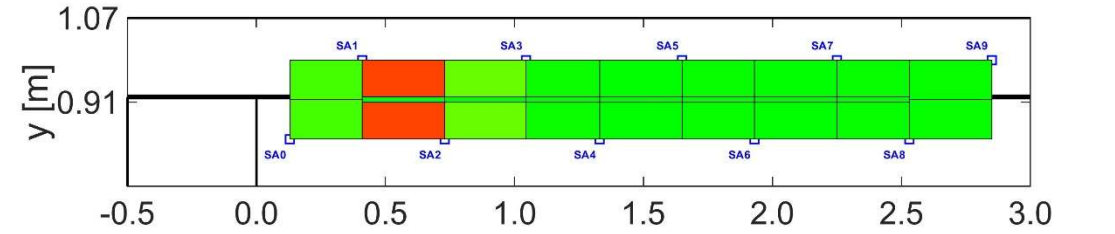


(e)- 150[kN]



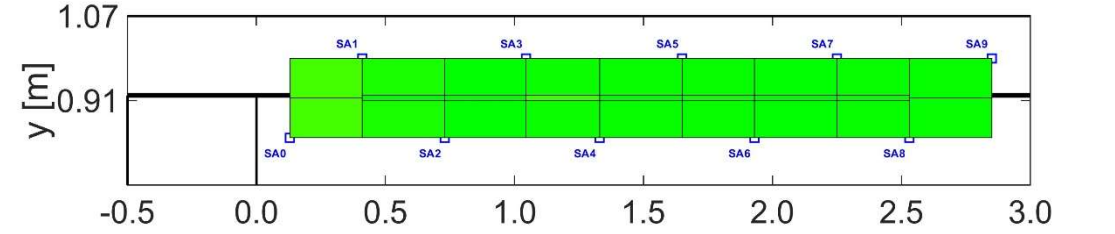


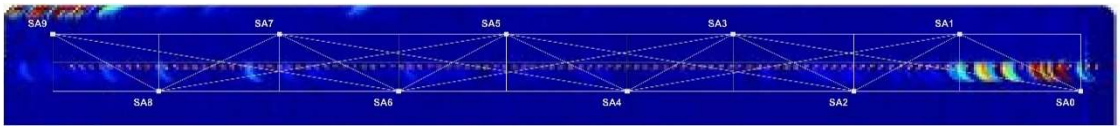
(f)- 200[kN]



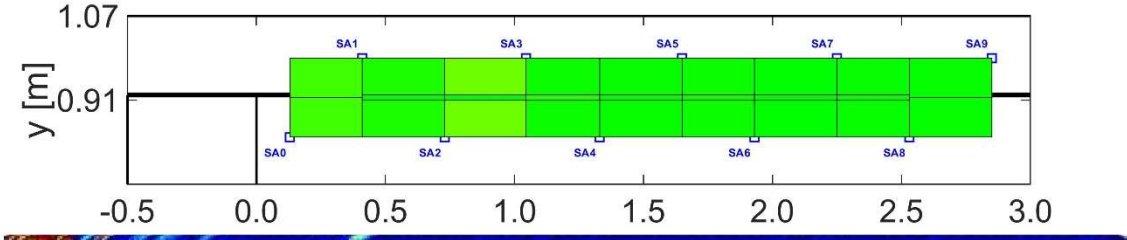
(no DIC Figure)

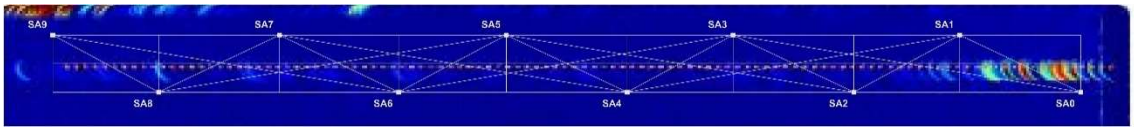
(g)- 250[kN]



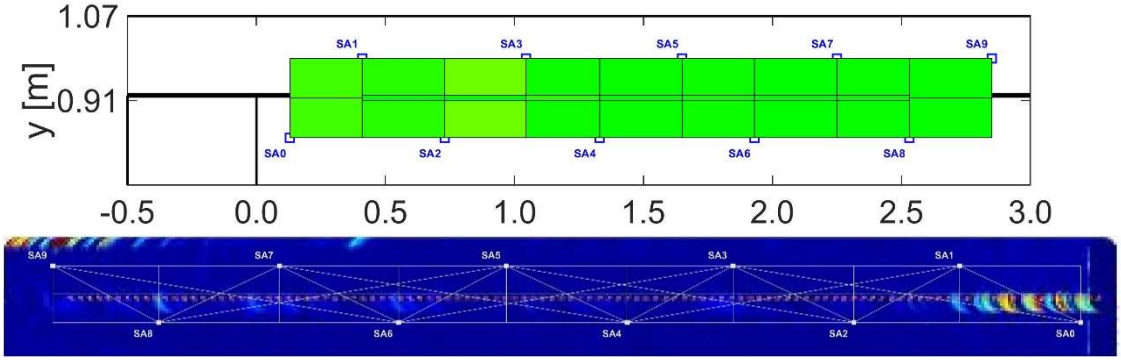


(h)- 300[kN]

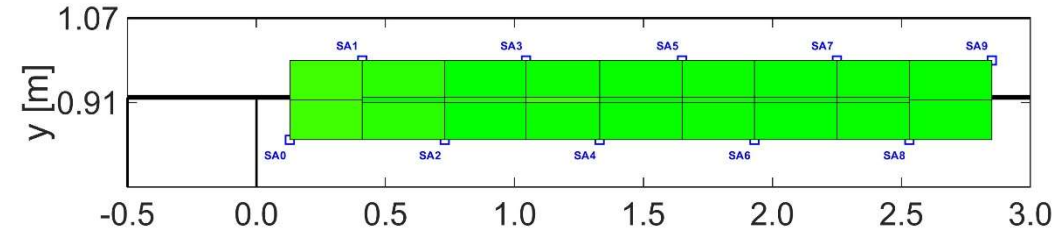




(i)- 350[kN]

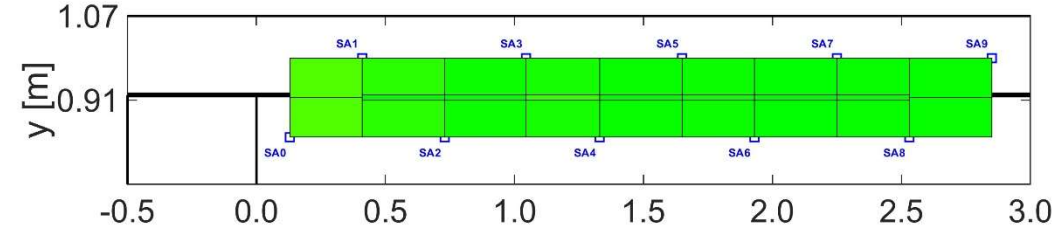


(j)- 400[kN]



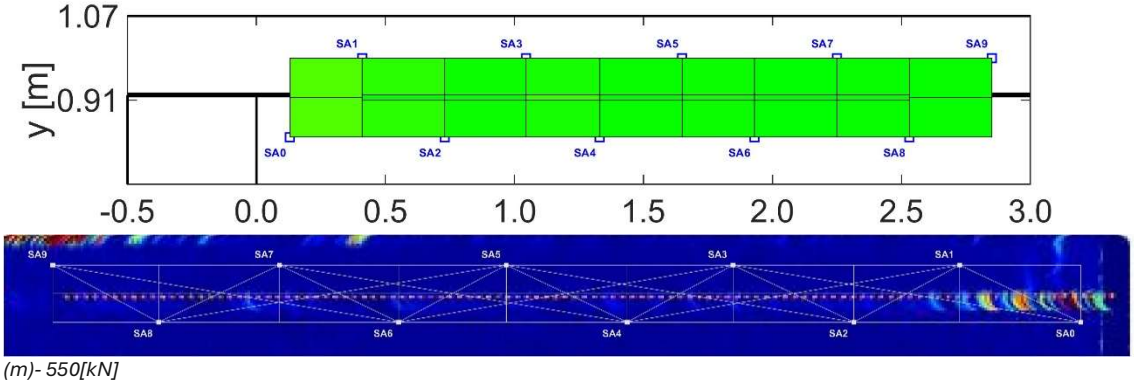
(no DIC Figure)

(k)- 450[kN]

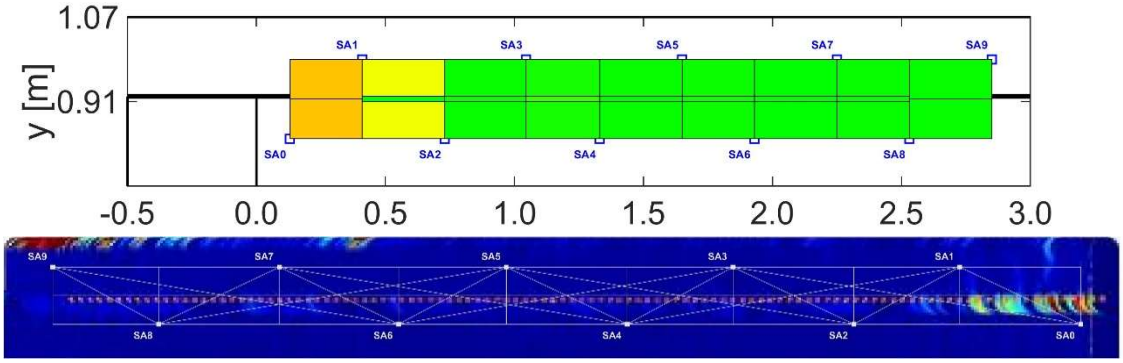


(no DIC Figure)

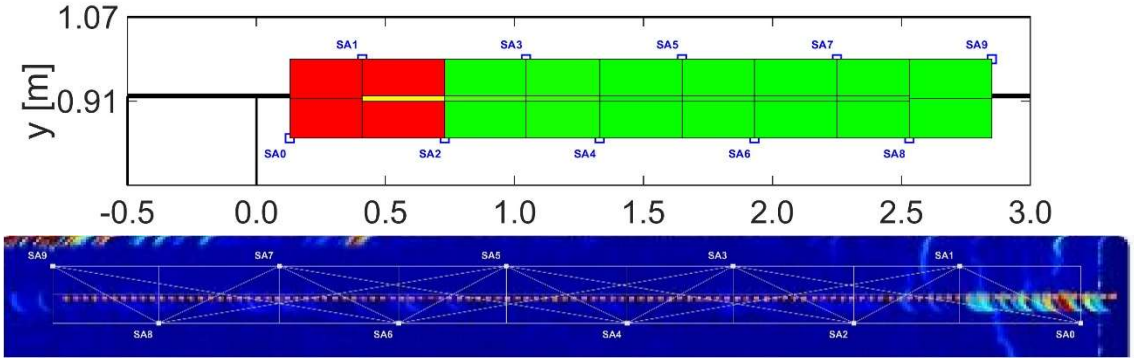
(l)- 500[kN]



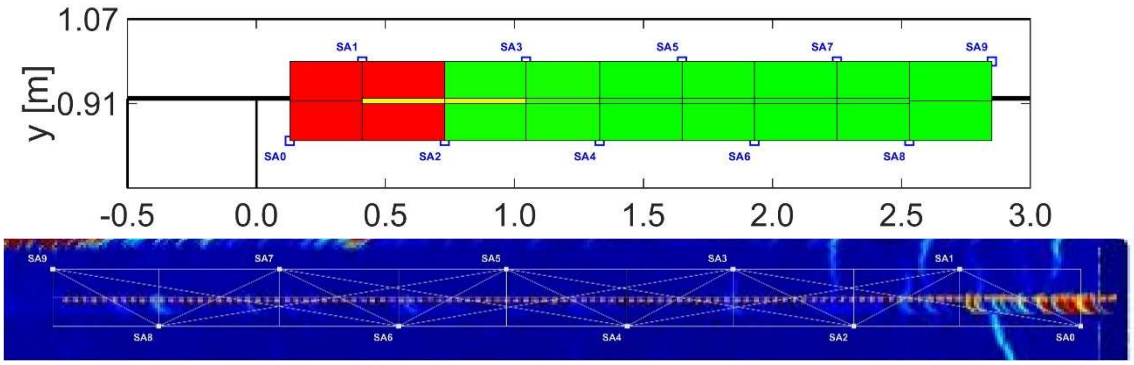
(m)- 550[kN]



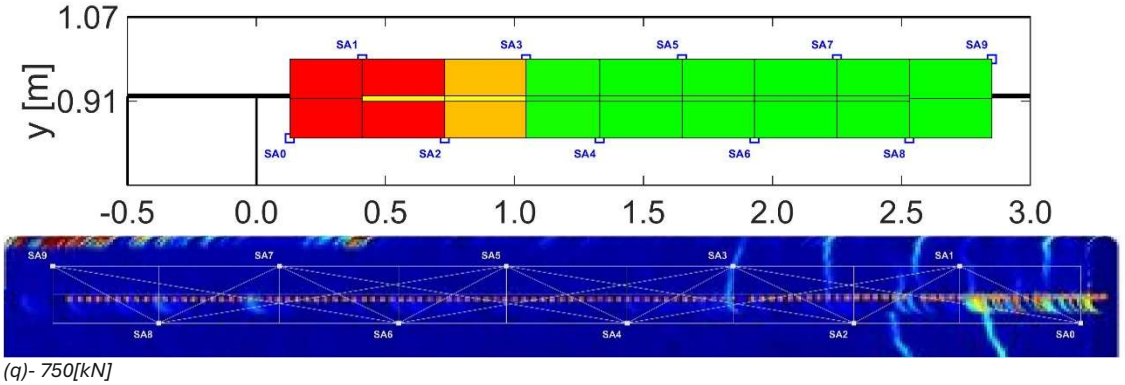
(n)- 600[kN]



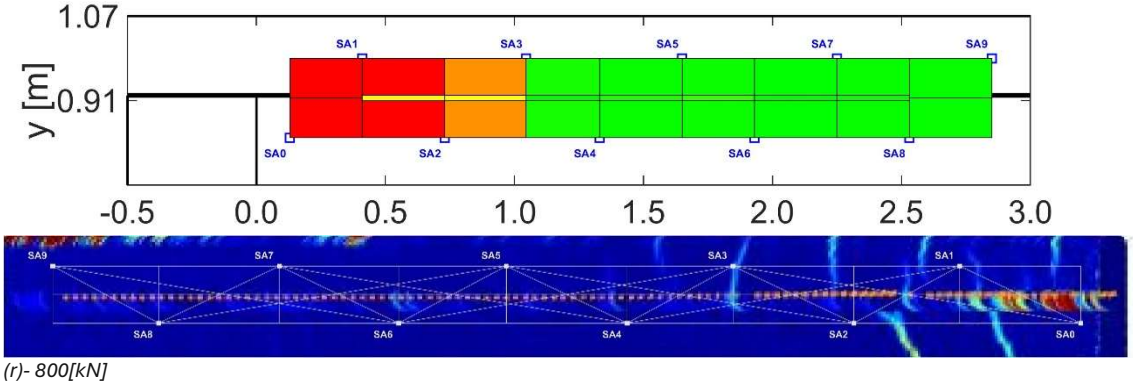
(o)- 650[kN]



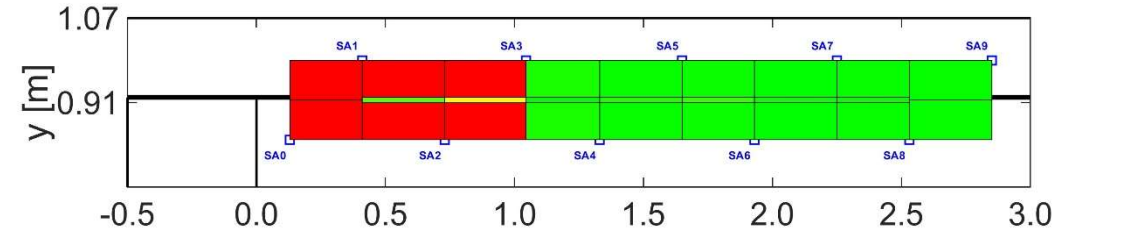
(p)- 700[kN]

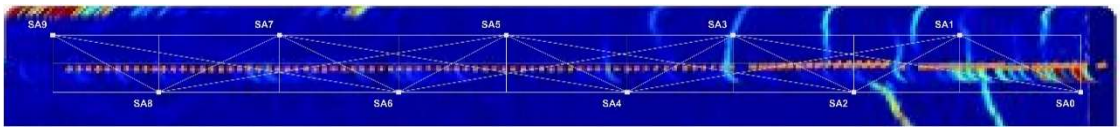


(q)- 750[kN]

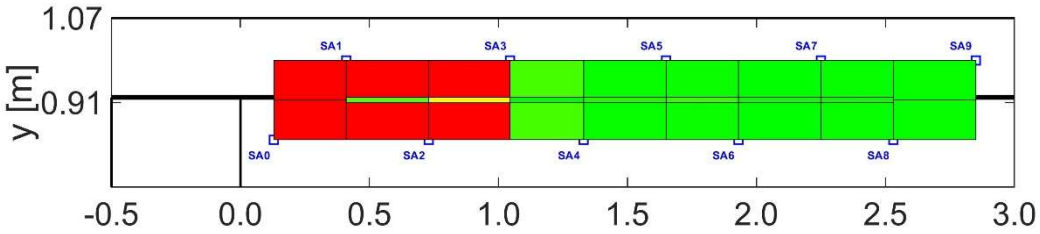


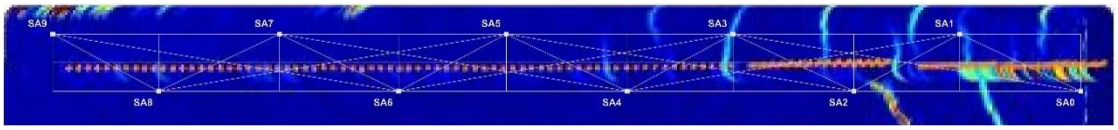
(r)- 800[kN]



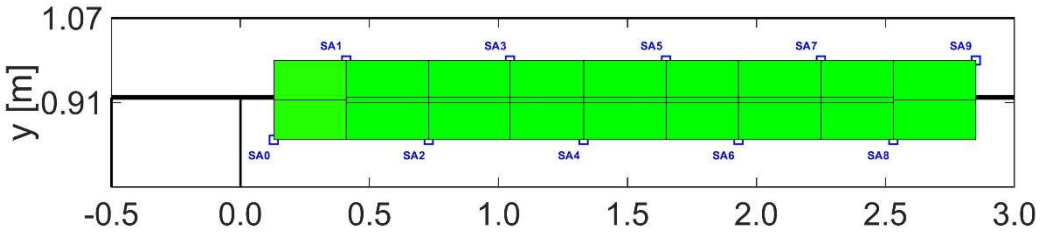


(s)- 850[kN]





(t)- 900[kN]

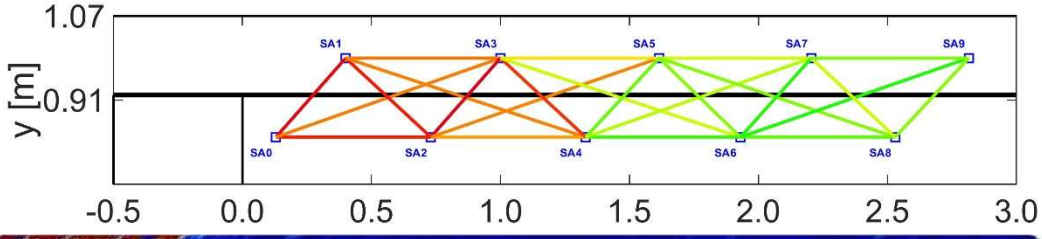


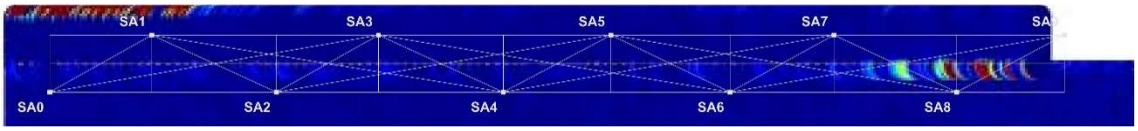
(no DIC Figure)
(u)- 0[kN]

Appendix V.3 – beam S10H1D

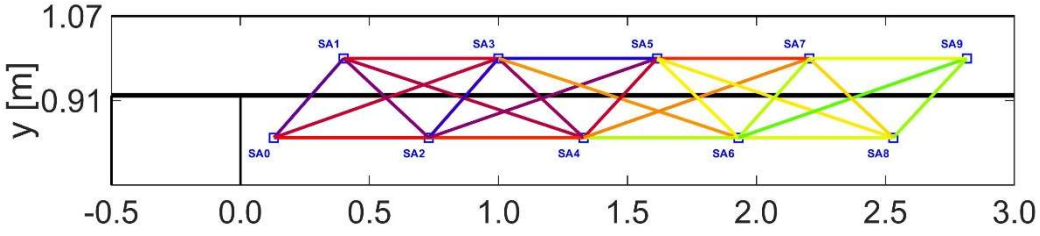
CWI – Relative Velocity Change (ε) vs DIC

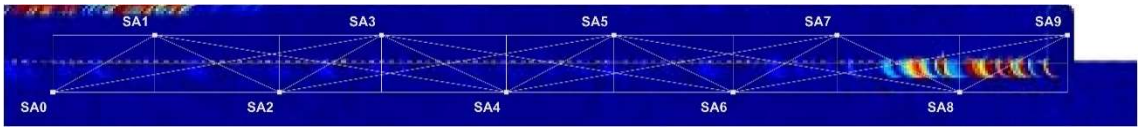
(b) – (p)



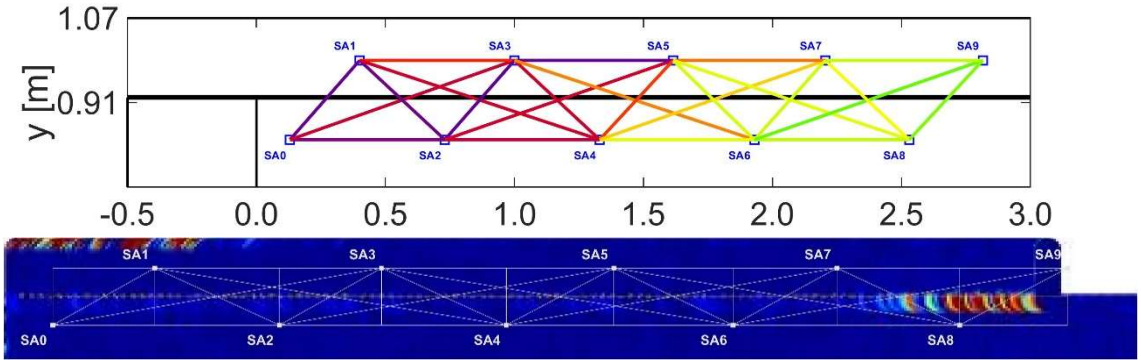


(b)- 50[kN]

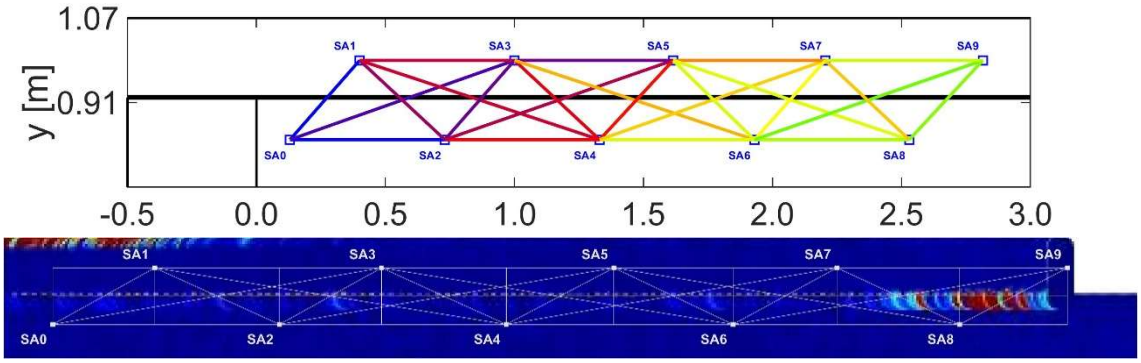




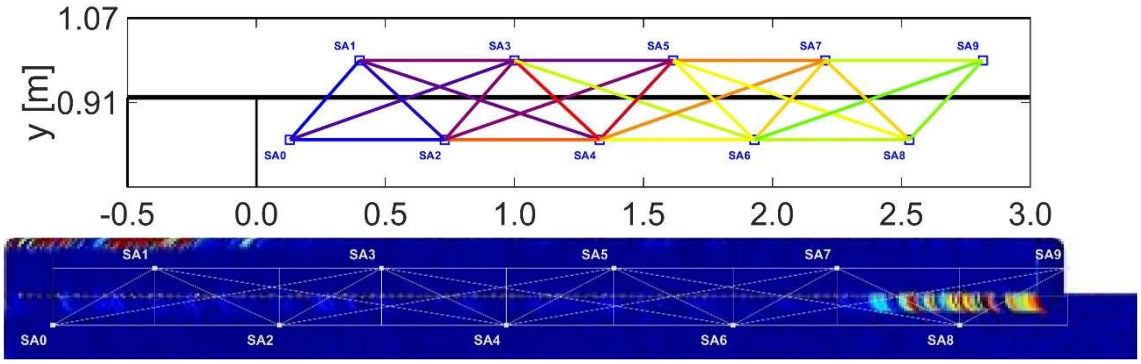
(c)- 100[kN]



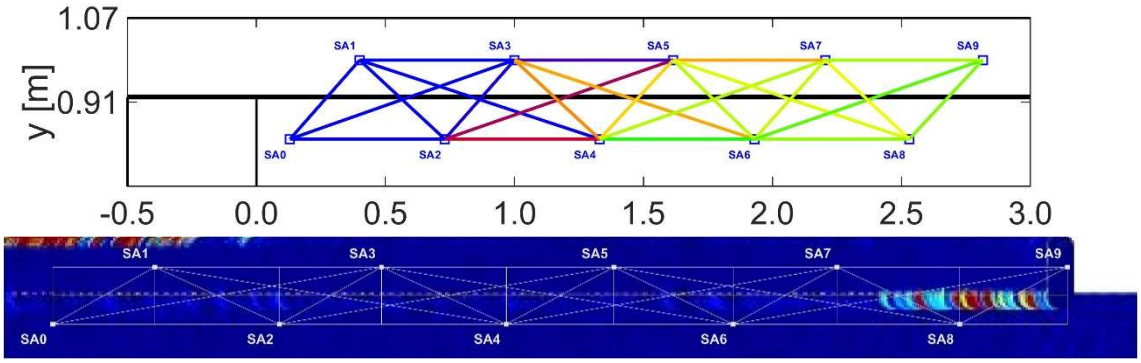
(d)- 150[kN]



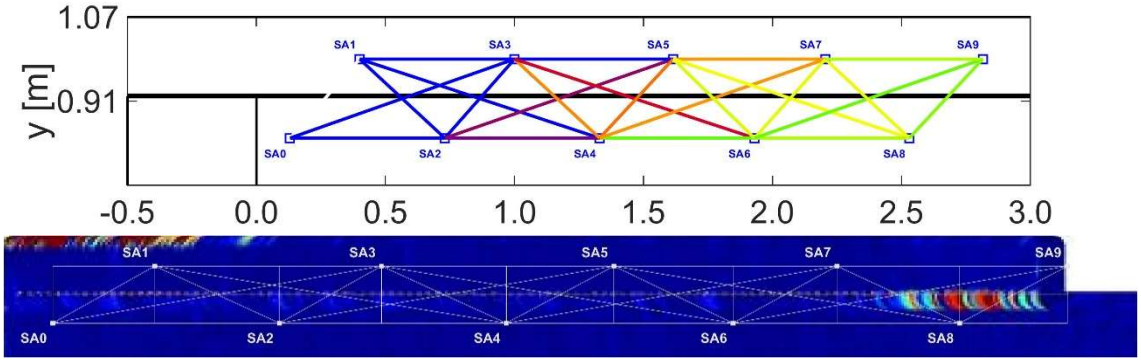
(e)- 200[kN]



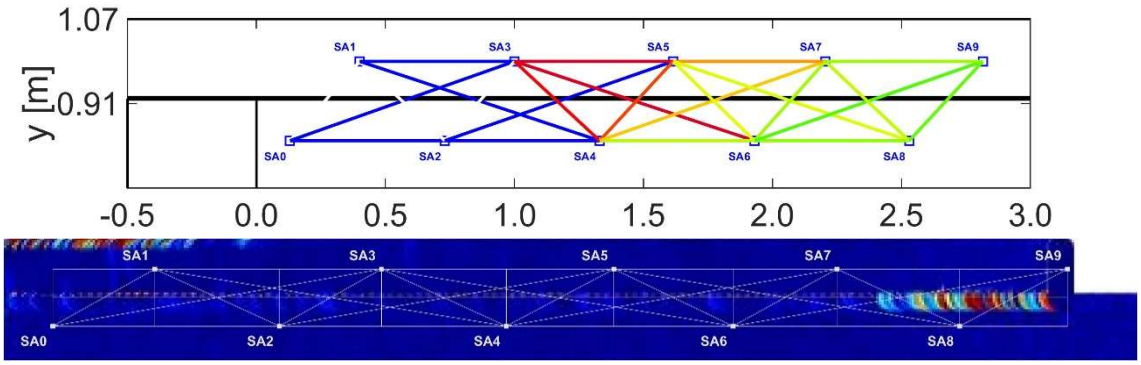
(f)- 250[kN]



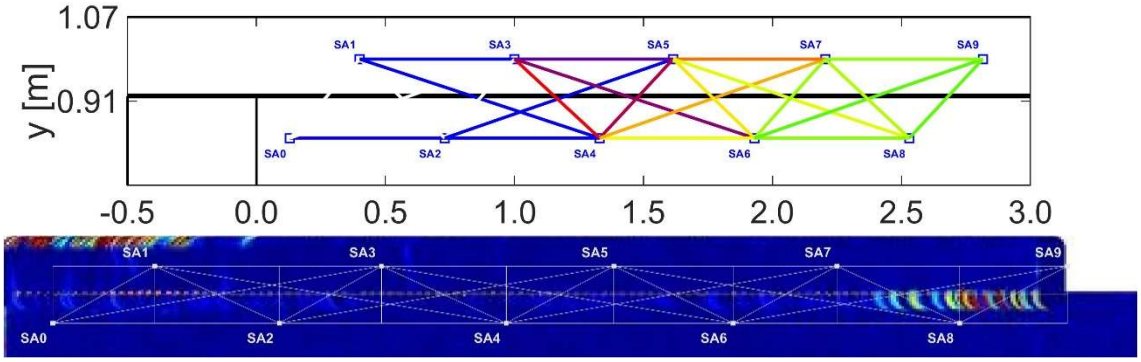
(g)- 300[kN]



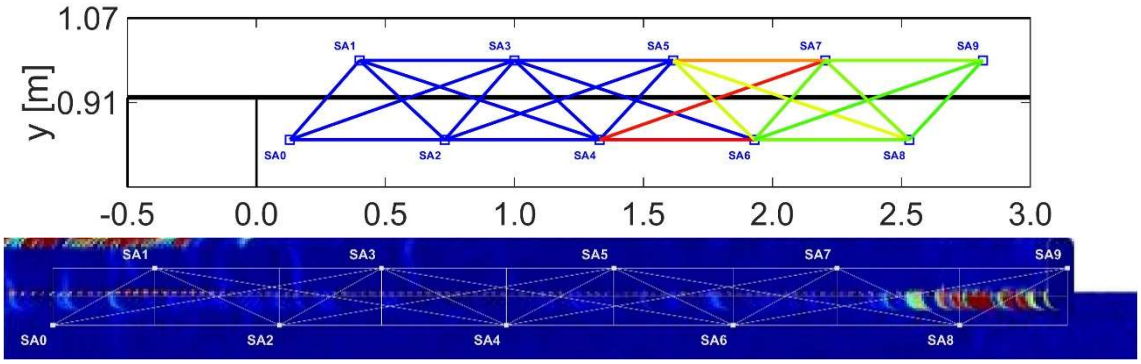
(h)- 350[kN]



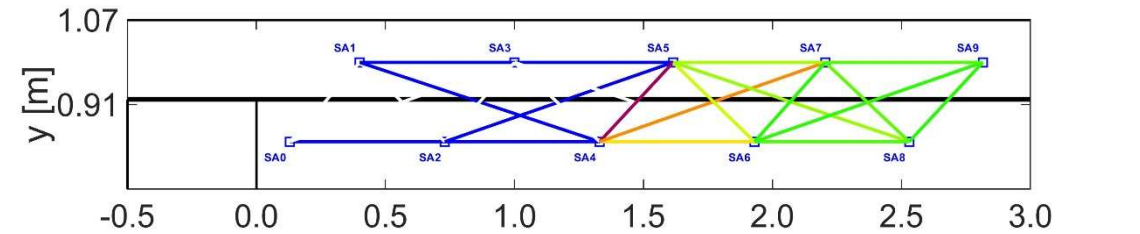
(i)- 400[kN]

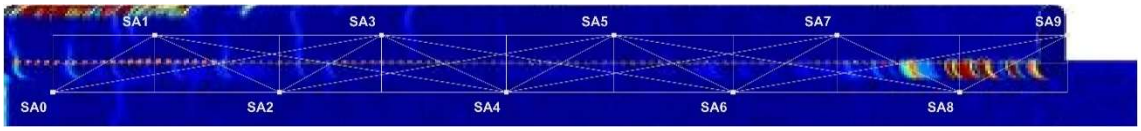


(j)- 450[kN]

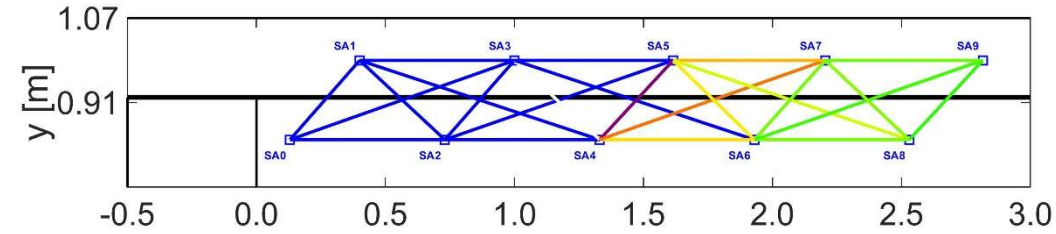


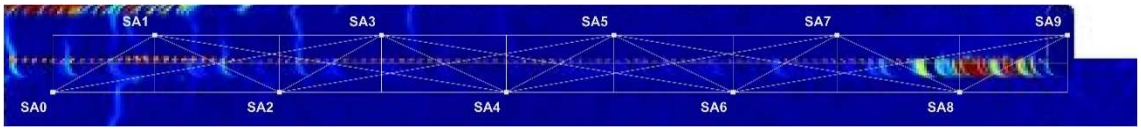
(k)- 500[kN]



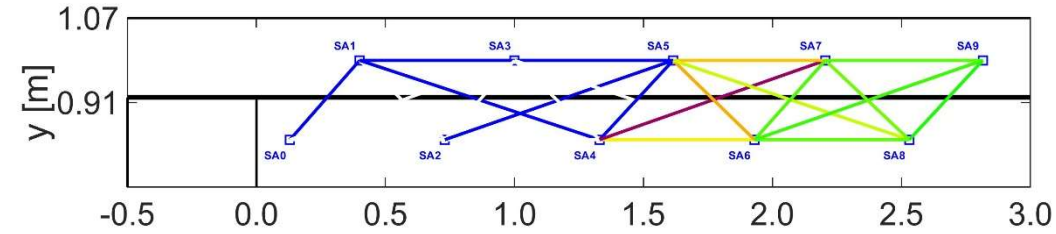


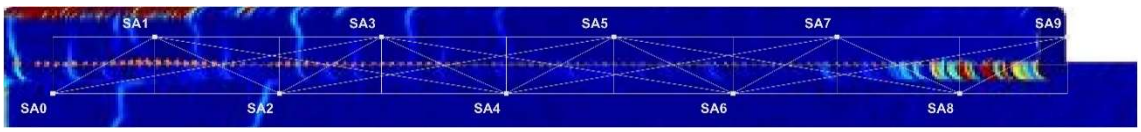
(l)- 550[kN]



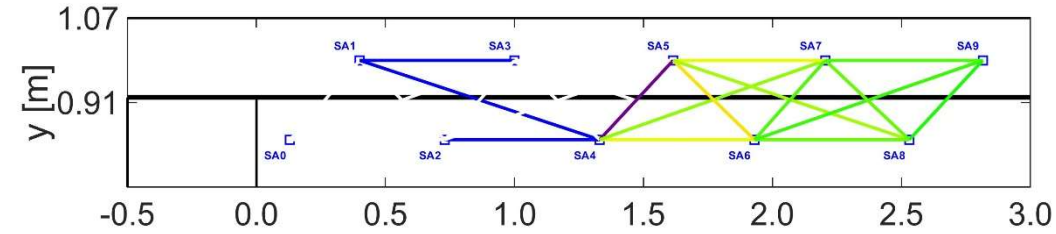


(m)- 600[kN]



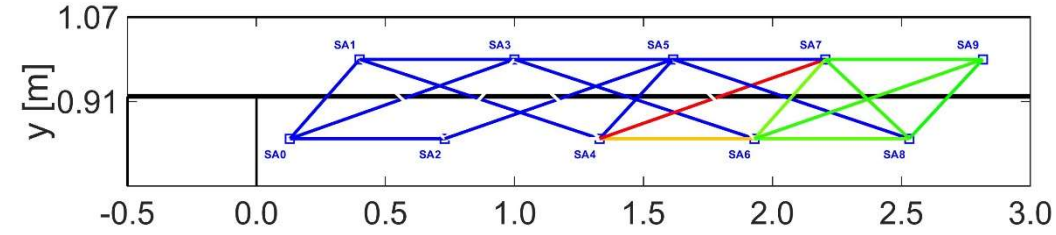


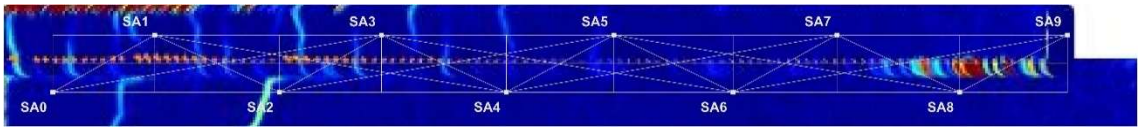
(n)- 650[kN]



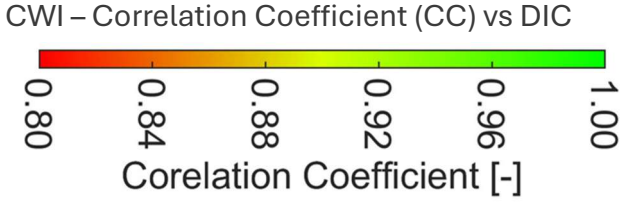
(no DIC Figure)

(o)- 700[kN]

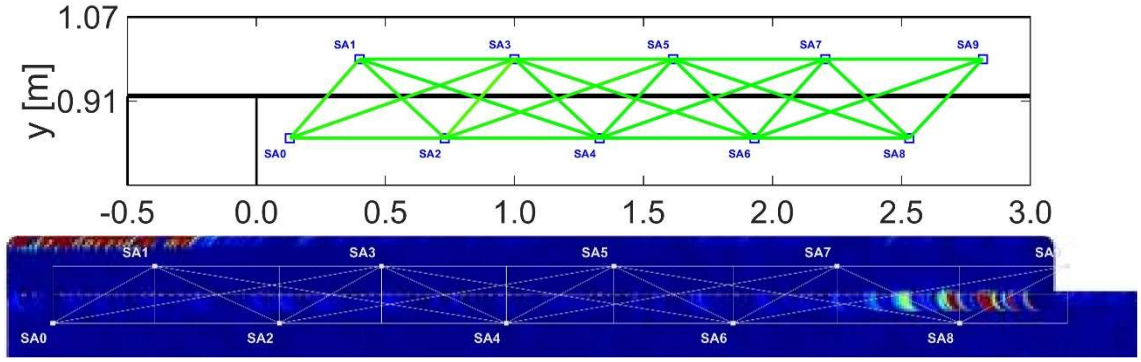




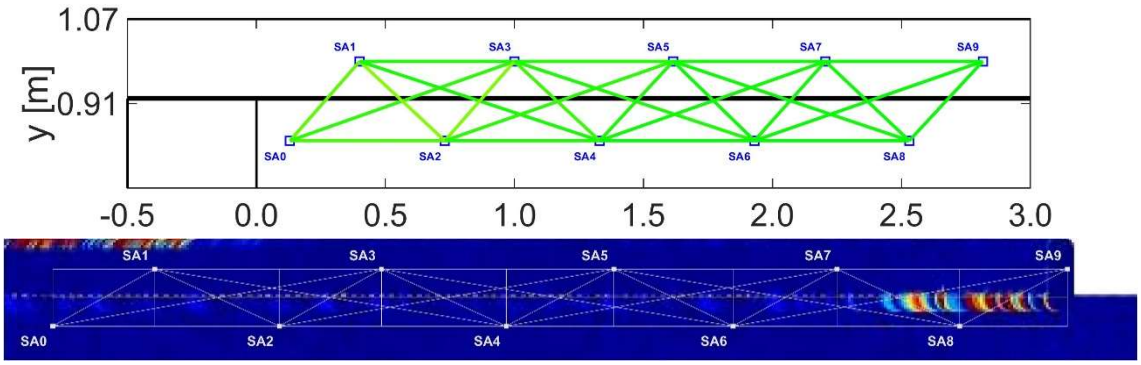
(p)- 750[kN]



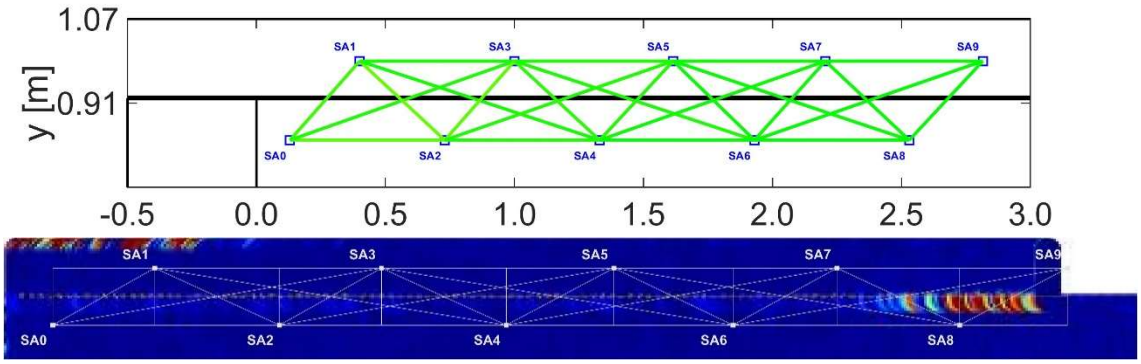
(b)-(p)



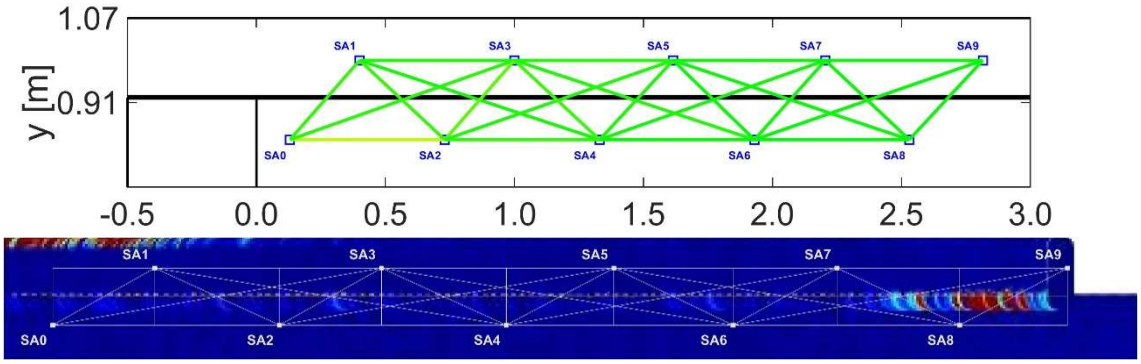
(b)- 50[kN]



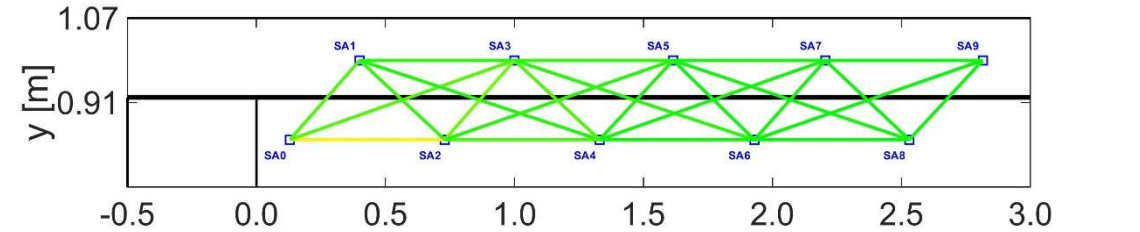
(c)- 100[kN]

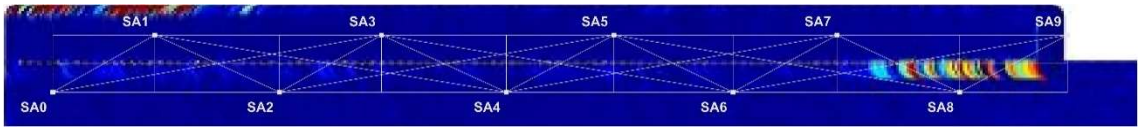


(d)- 150[kN]

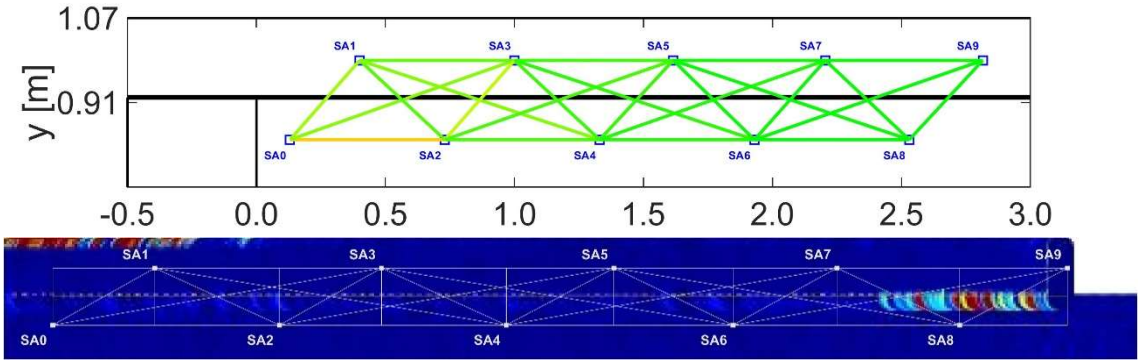


(e)- 200[kN]

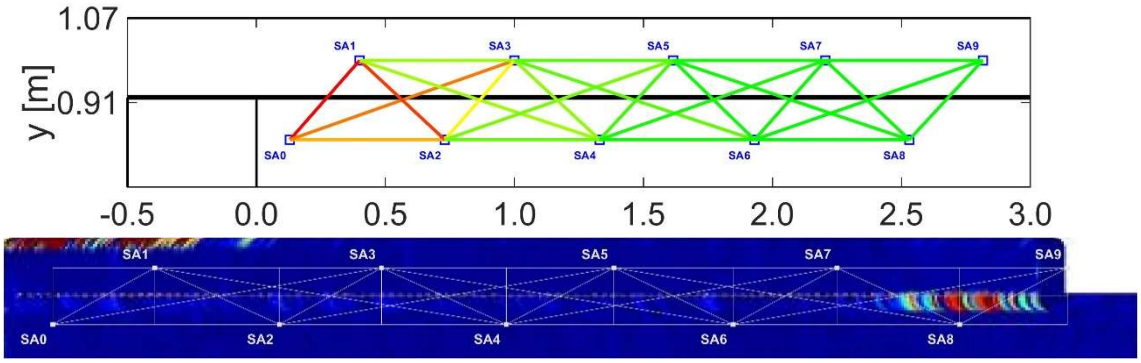




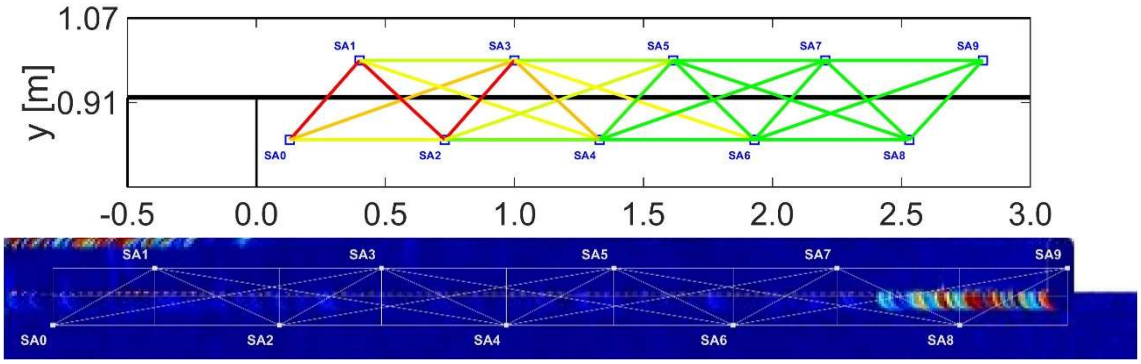
(f)- 250[kN]



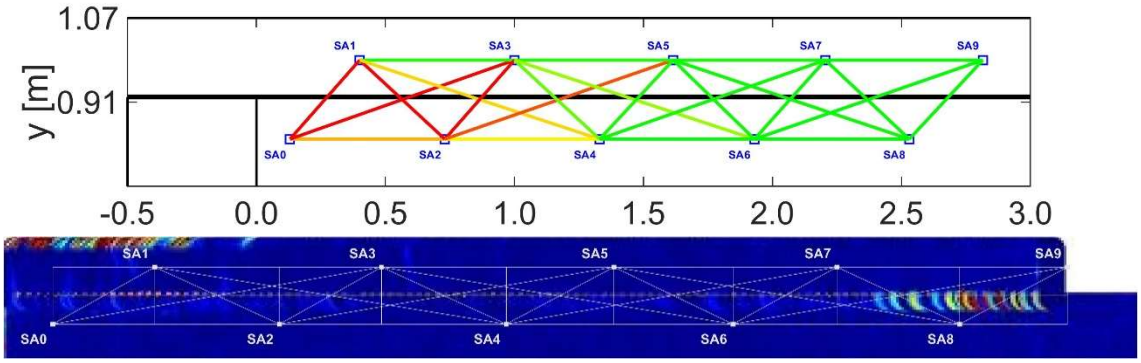
(g)- 300[kN]



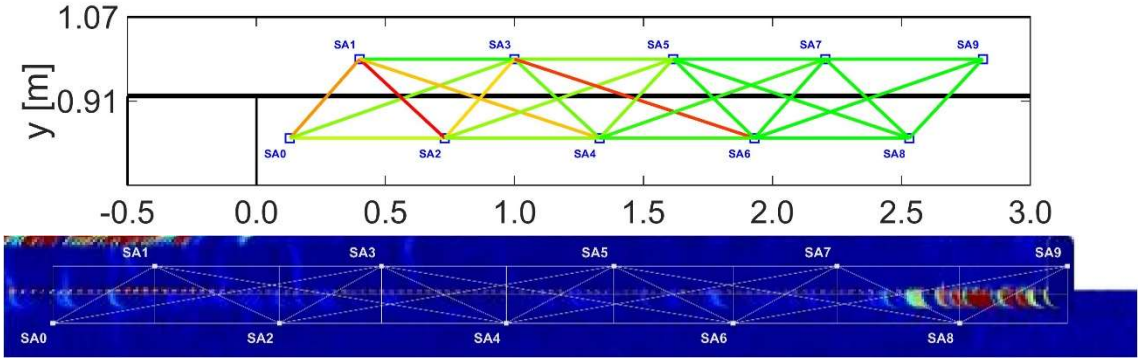
(h)- 350[kN]



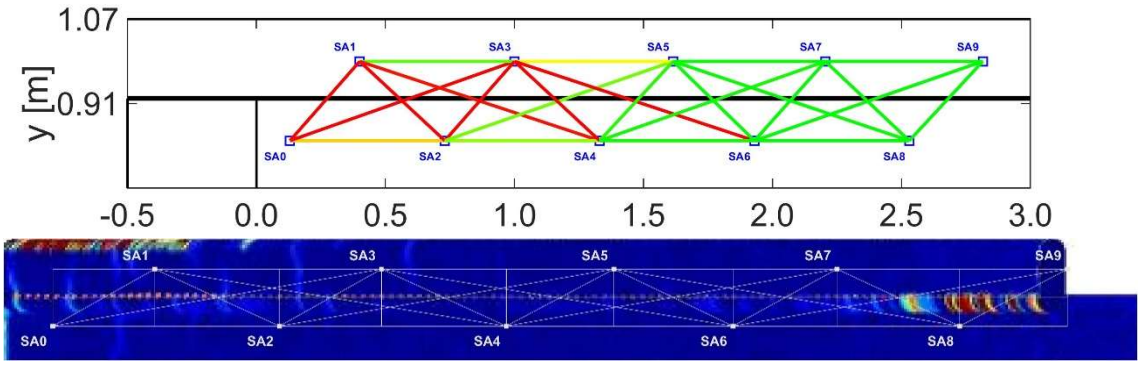
(i)- 400[kN]



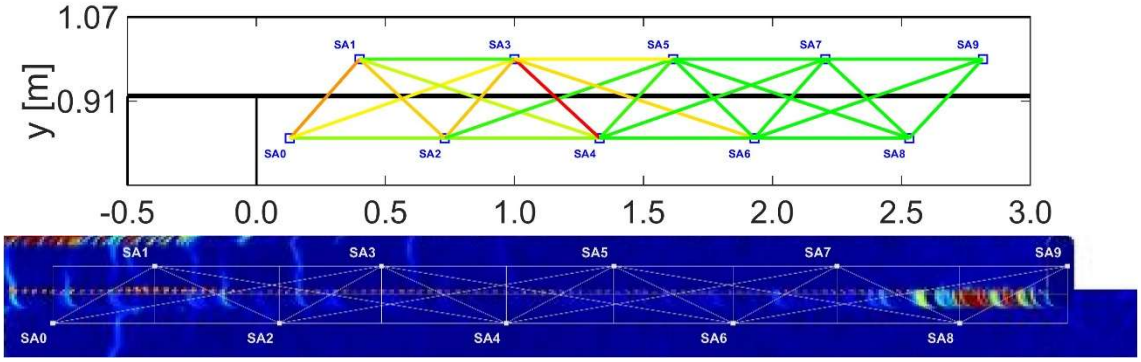
(j)- 450[kN]



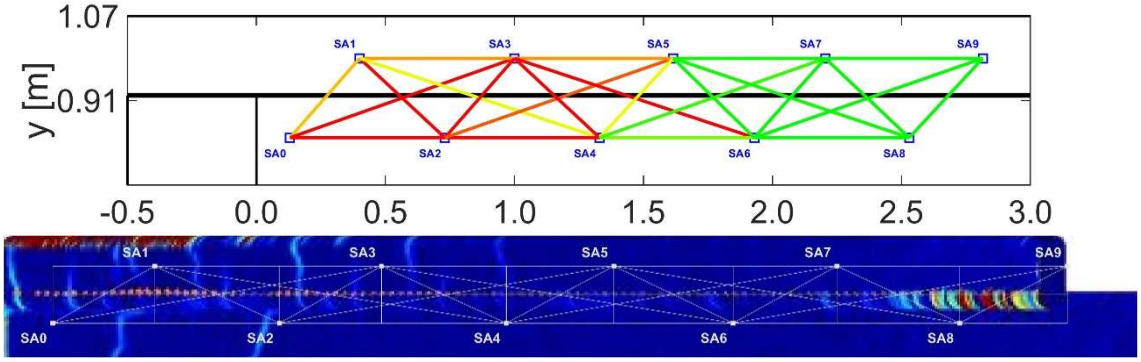
(k)- 500[kN]



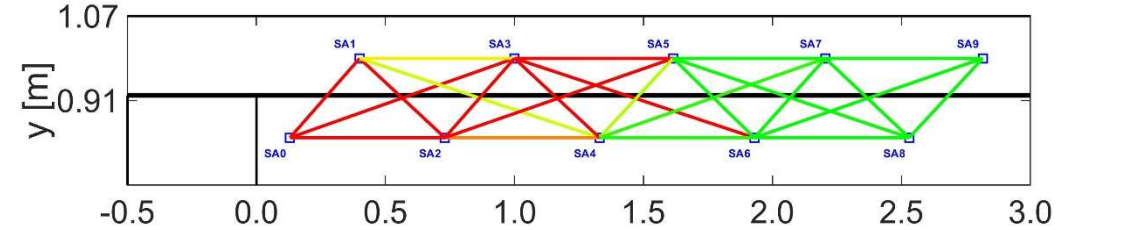
(l)- 550[kN]



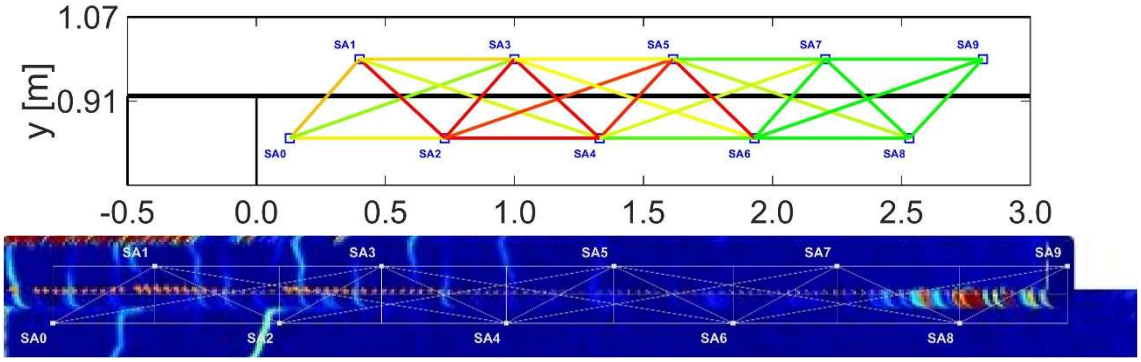
(m)- 600[kN]



(n)- 650[kN]

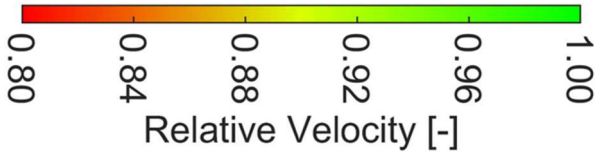


(no DIC Figure)
(o)- 700[kN]

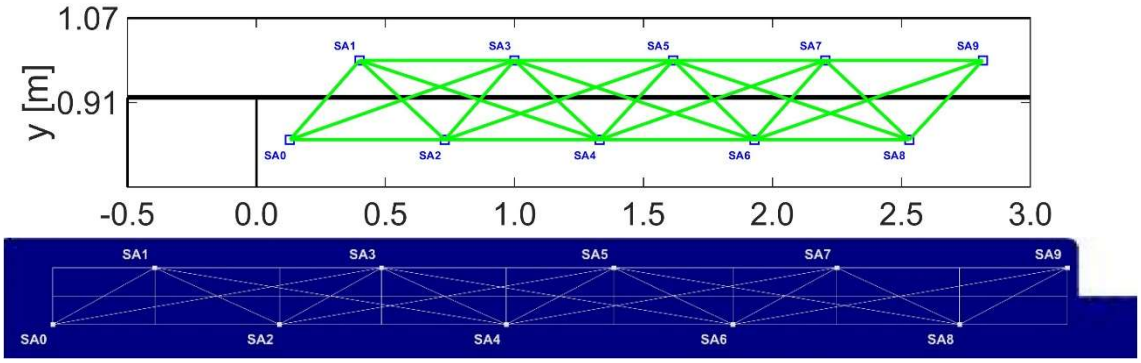


(p)- 750[kN]

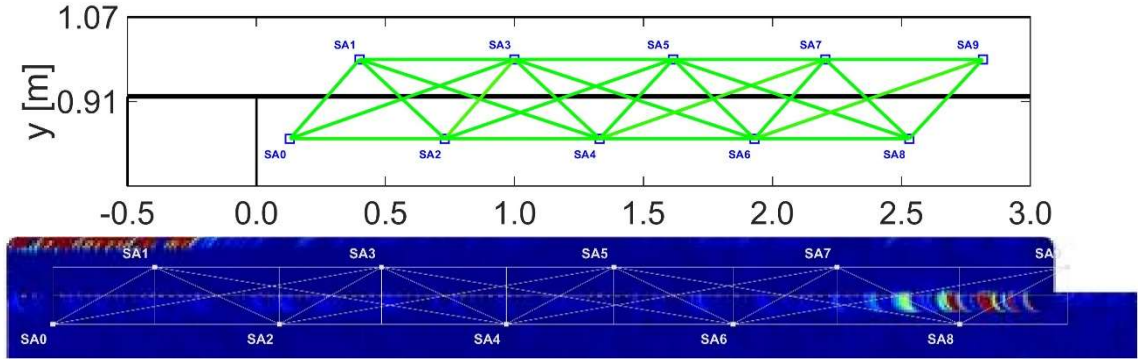
UPV – Relative Velocity vs DIC



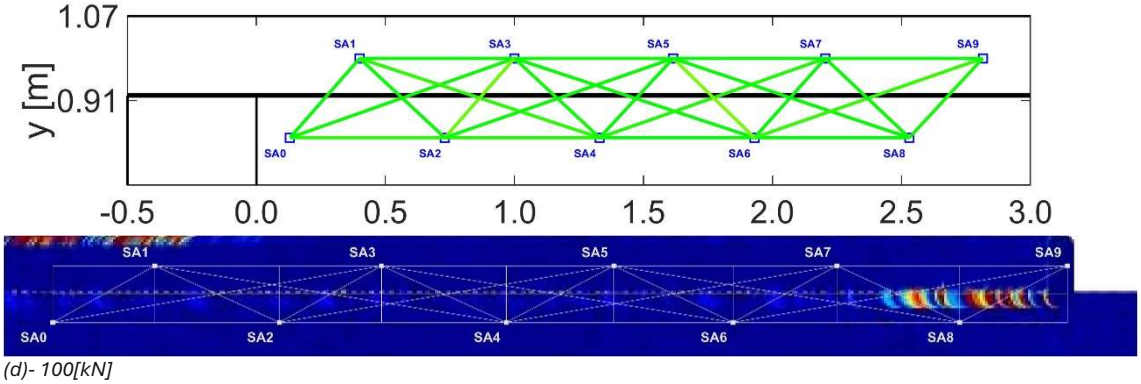
(b)-(ac)



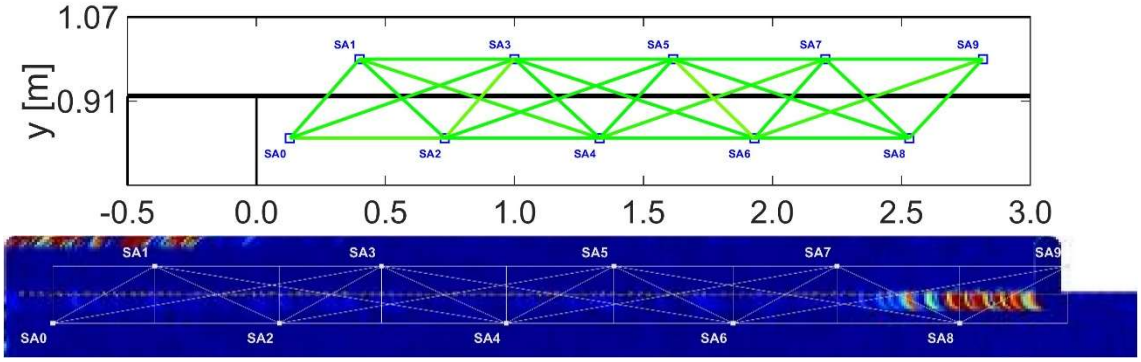
(b)- 0[kN]



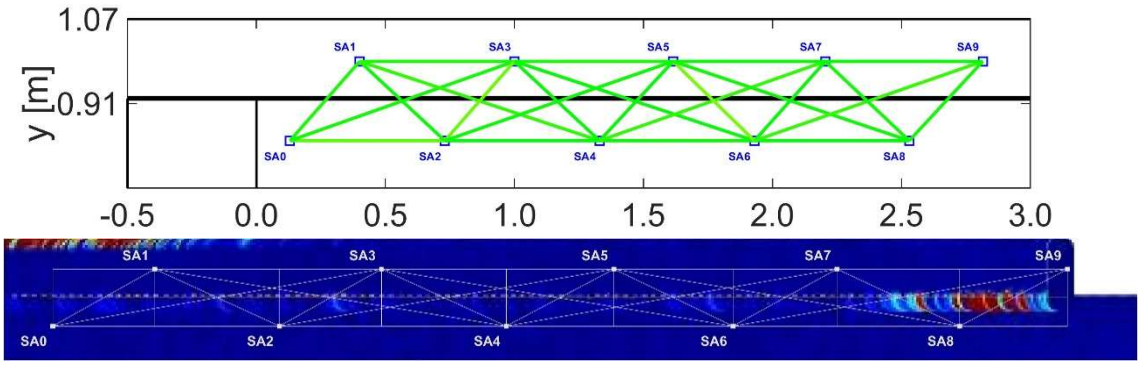
(c)- 50[kN]



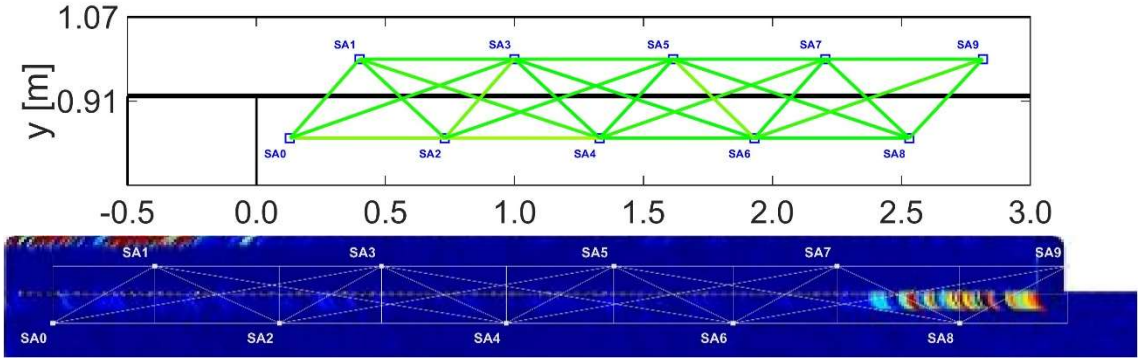
(d)- 100[kN]



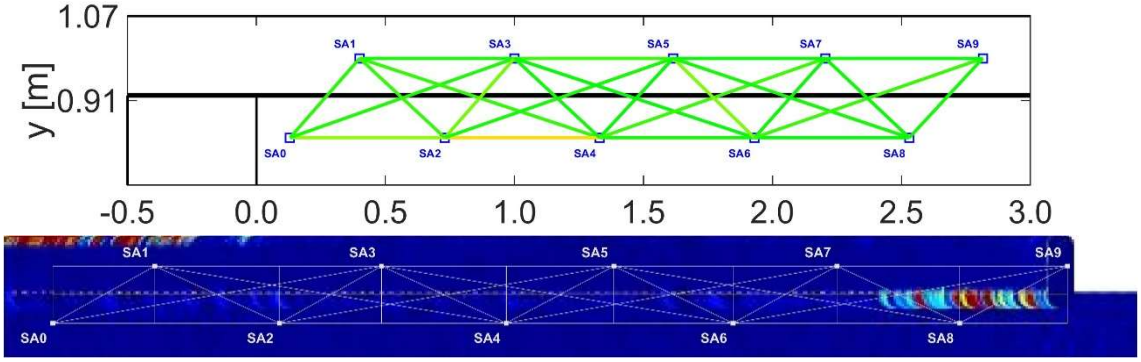
(e)- 150[kN]



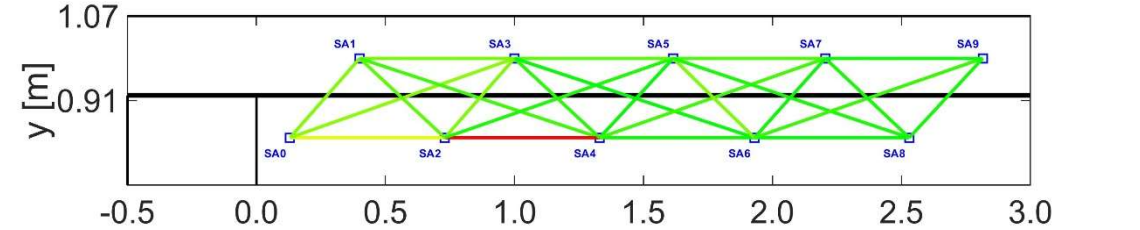
(f)- 200[kN]

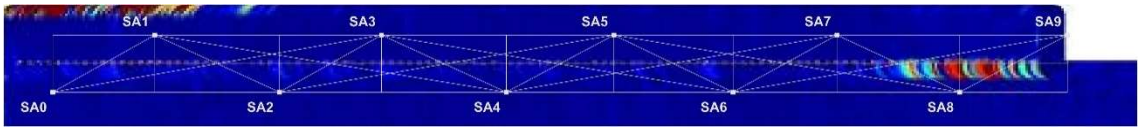


(g)- 250[kN]

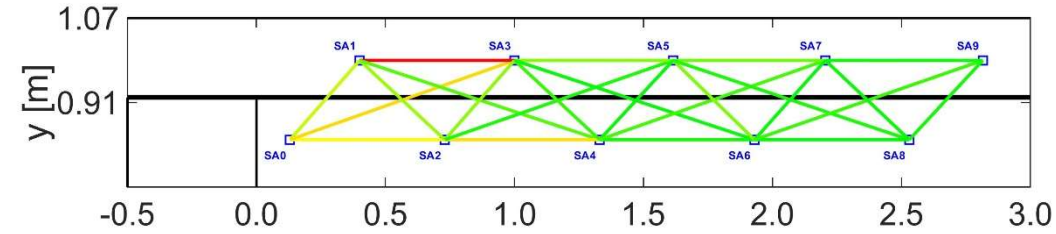


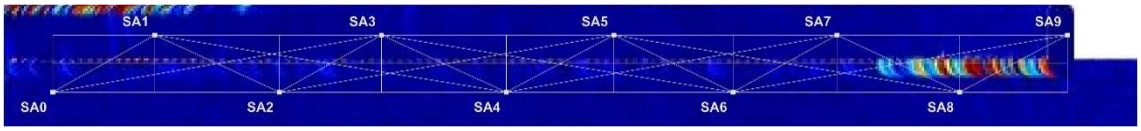
(h)- 300[kN]



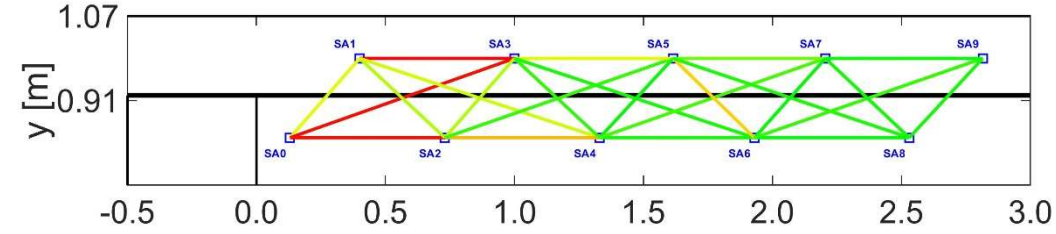


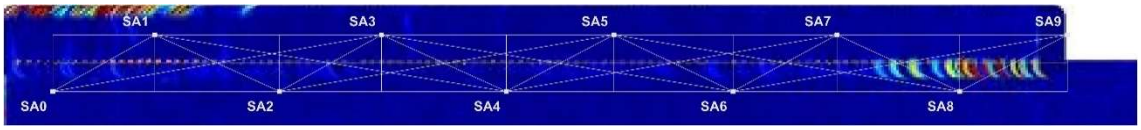
(i)- 350[kN]



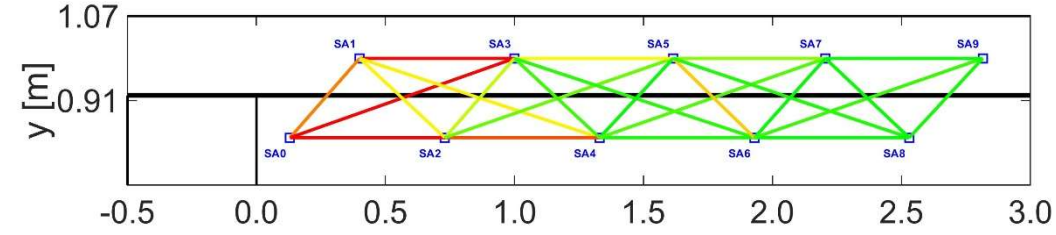


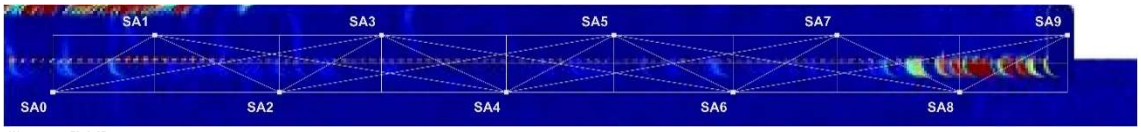
(j)- 400[kN]



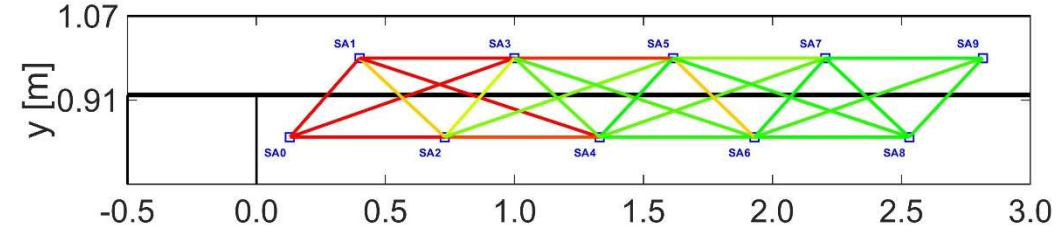


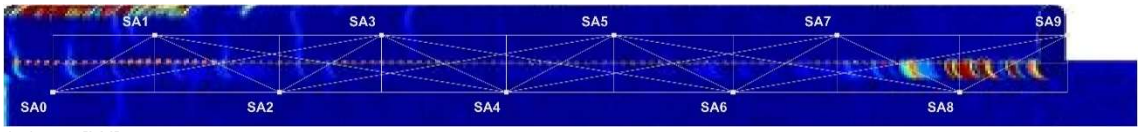
(k)- 450[kN]



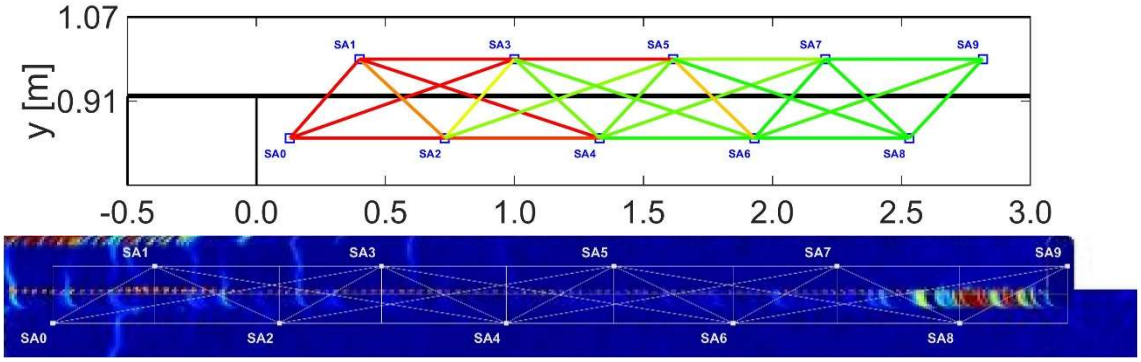


(l)- 500[kN]

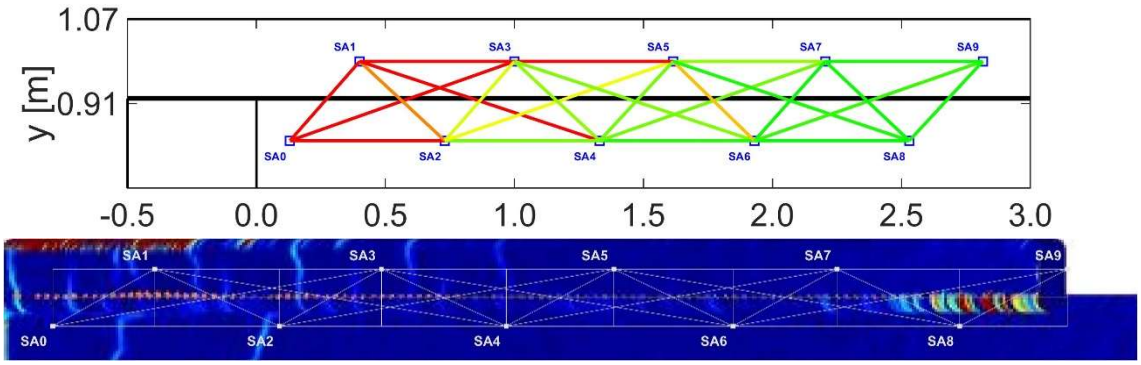




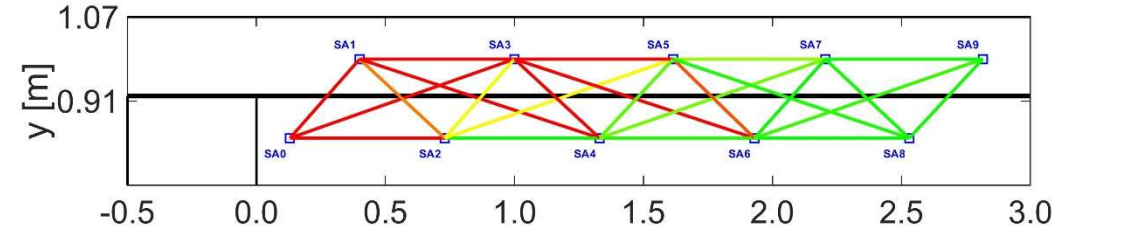
(m)- 550[kN]



(n)- 600[kN]

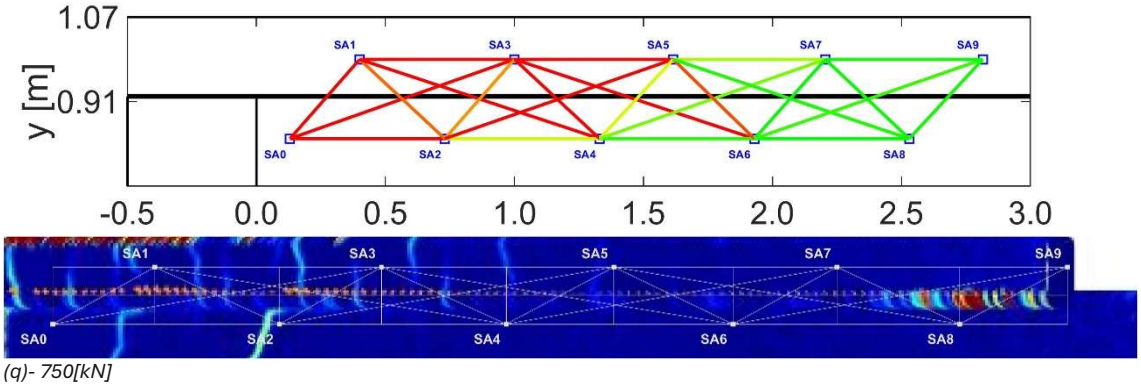


(o)- 650[kN]

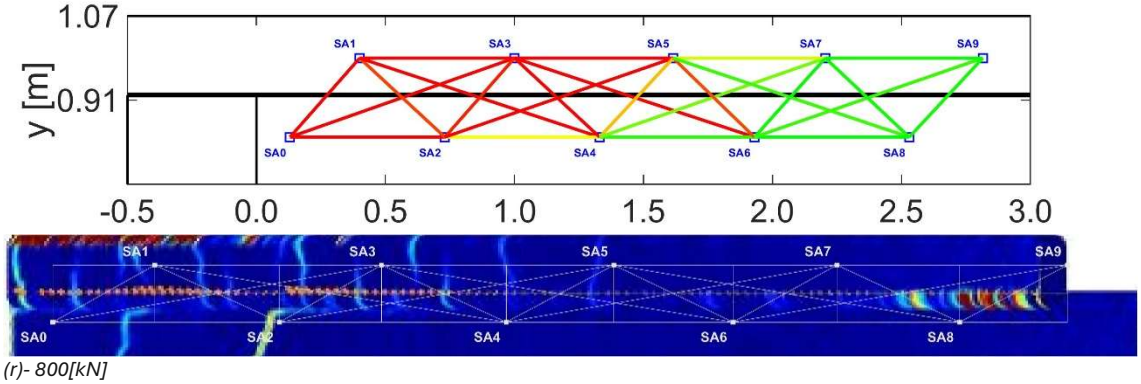


(no DIC Figure)

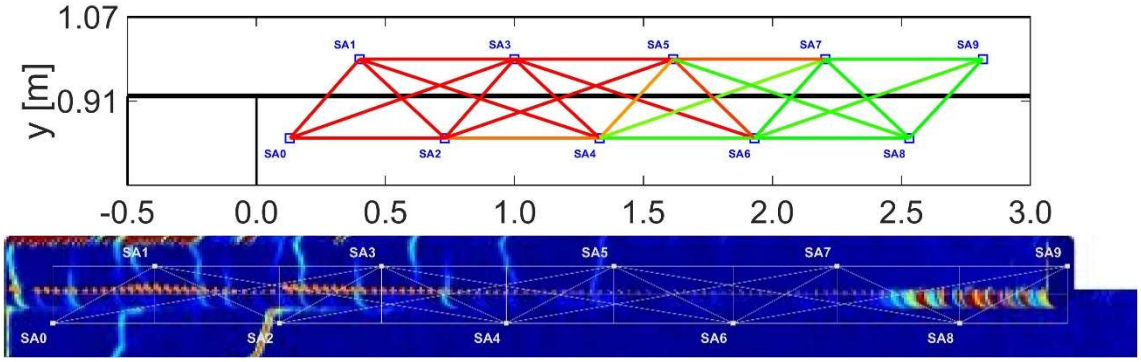
(p)- 700[kN]



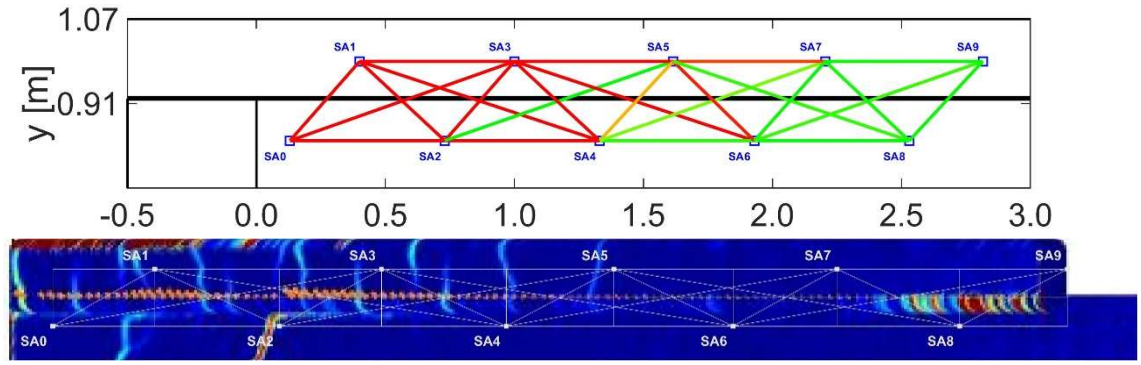
(q)- 750[kN]



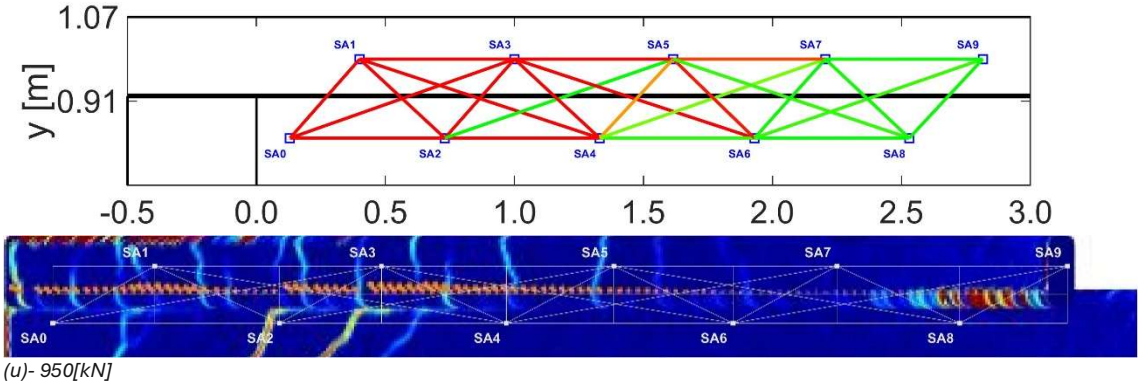
(r)- 800[kN]



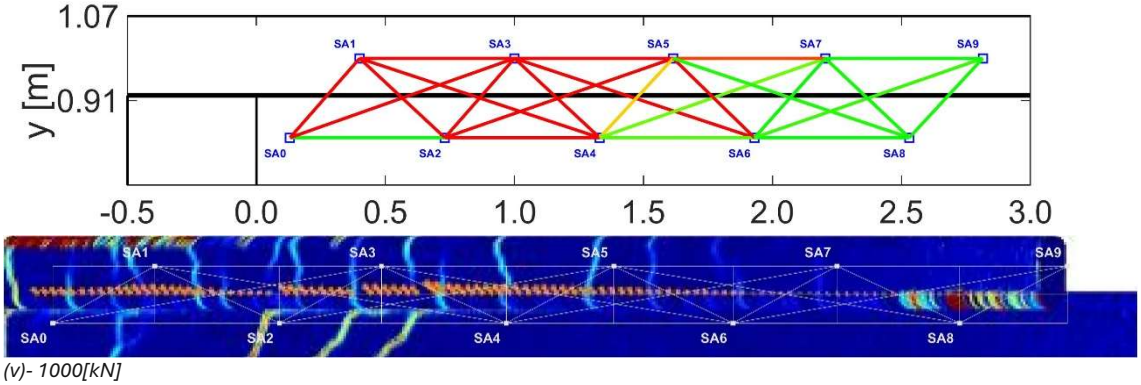
(s)- 850[kN]



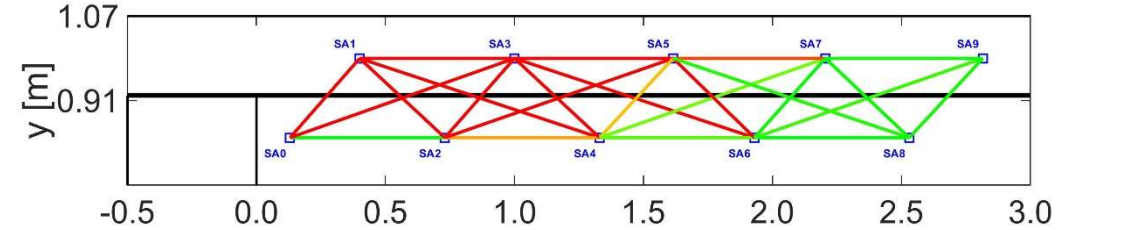
(t)- 900[kN]

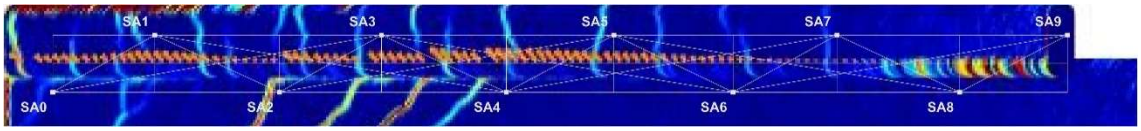


(u)- 950[kN]

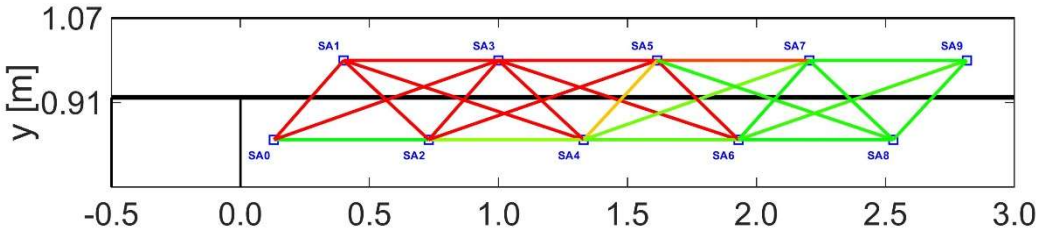


(v)- 1000[kN]



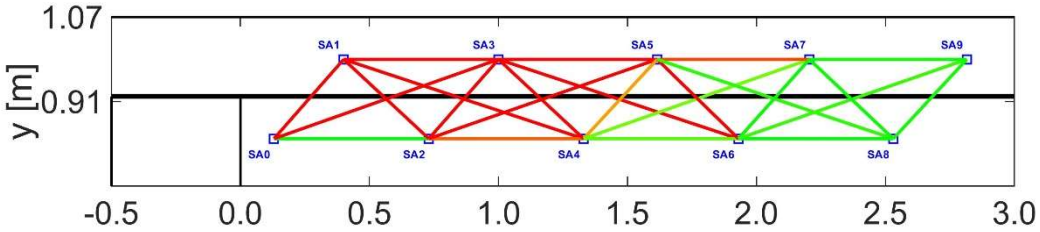


(w)- 1050[kN]



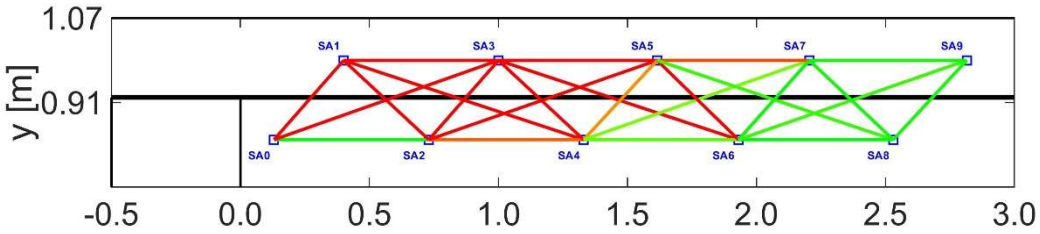
(no DIC Figure)

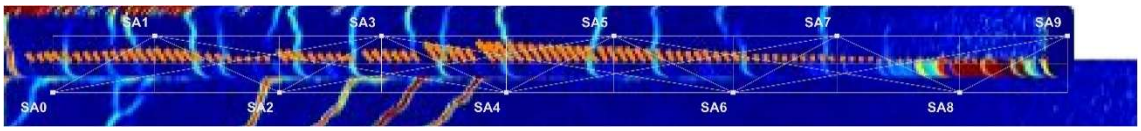
(x)- 1100[kN]



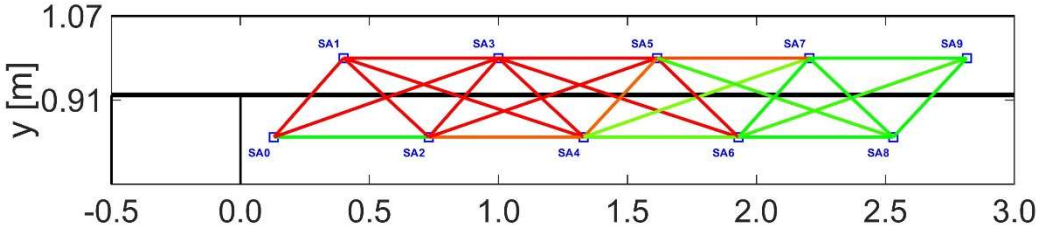
(no DIC Figure)

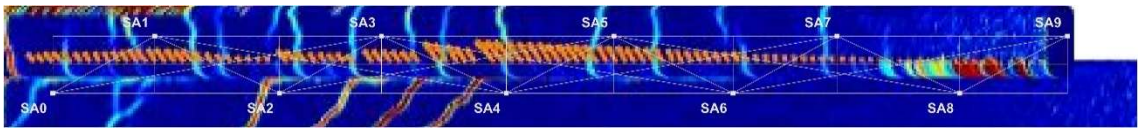
(y)- 1150[kN]



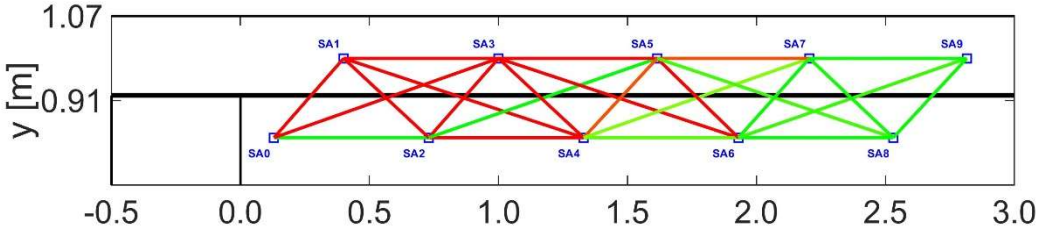


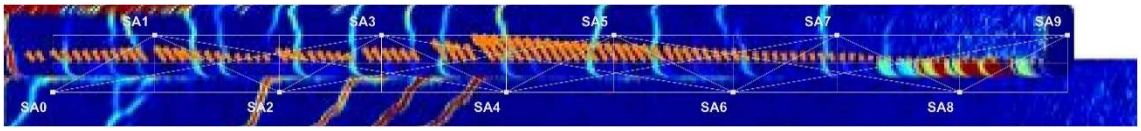
(z)- 1200[kN]



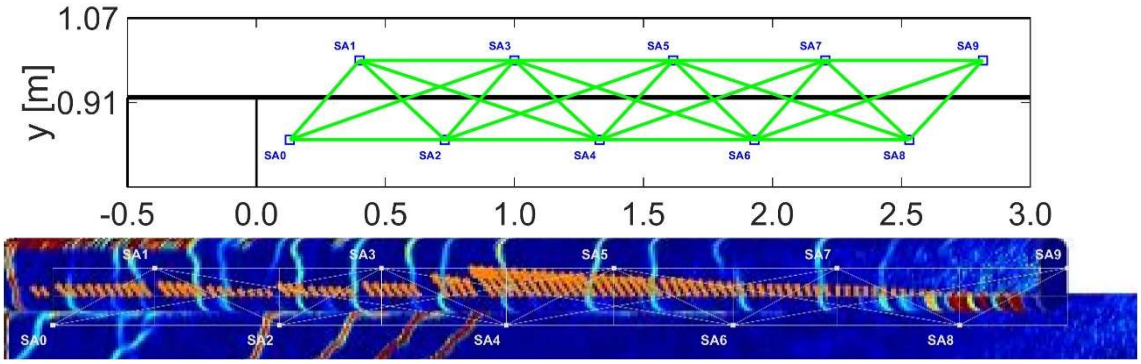


(aa)- 1250[kN]



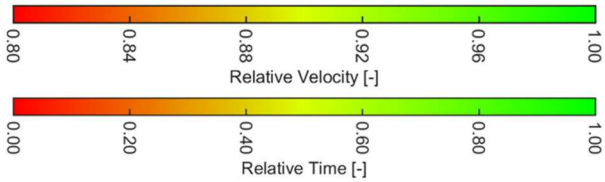


(ab)- 1300[kN]

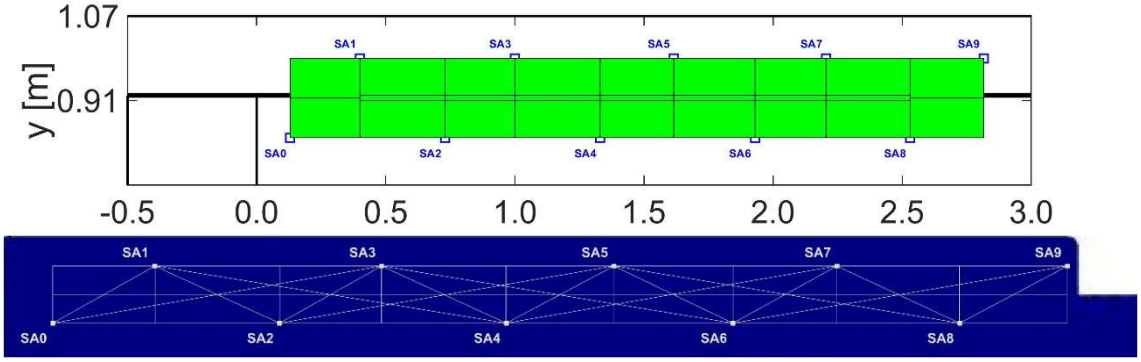


(ac)- 0[kN]

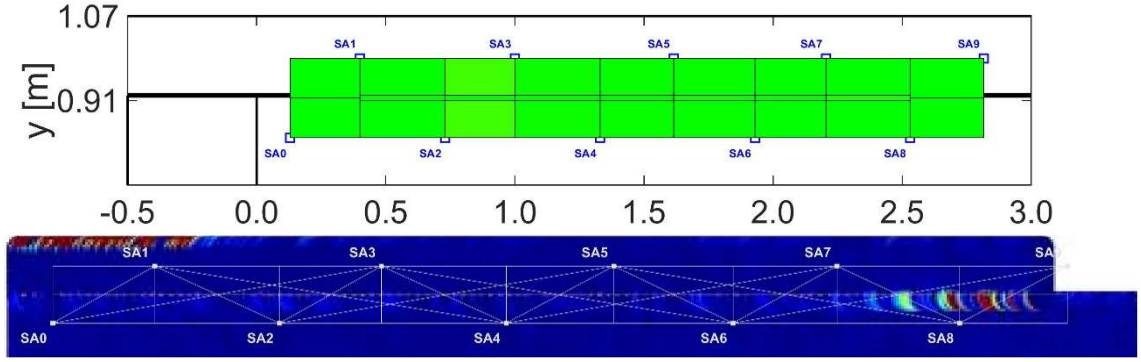
Tomography – Relative Field Velocity & Interface Time Interference (dt) vs DIC



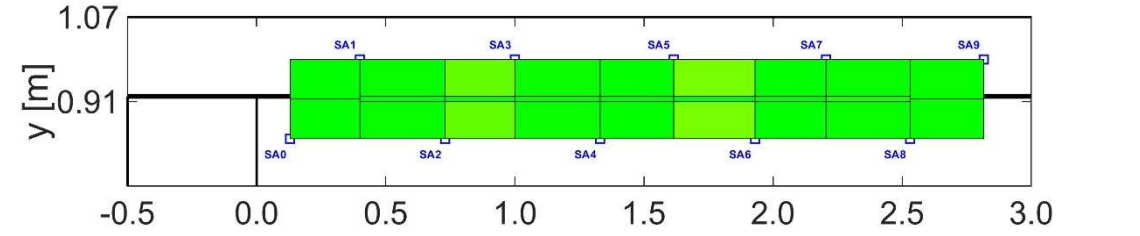
(b)-(ad)

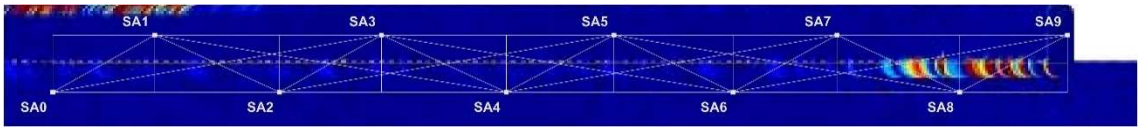


(b)- 0[kN]

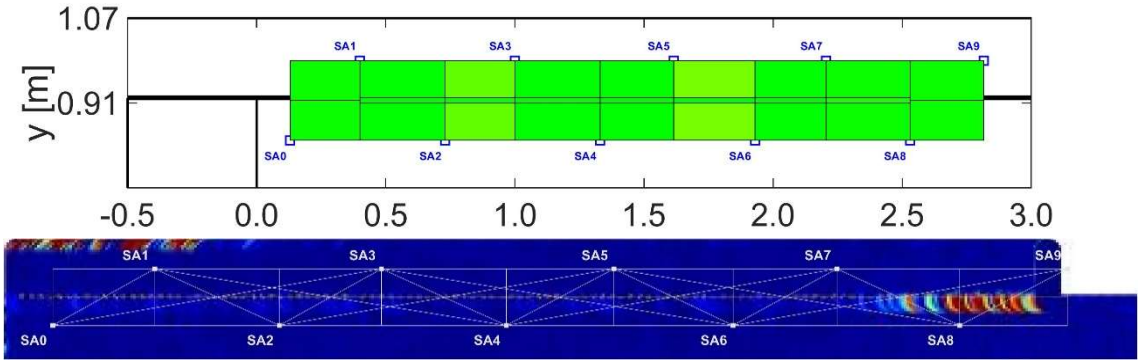


(c)- 50[kN]

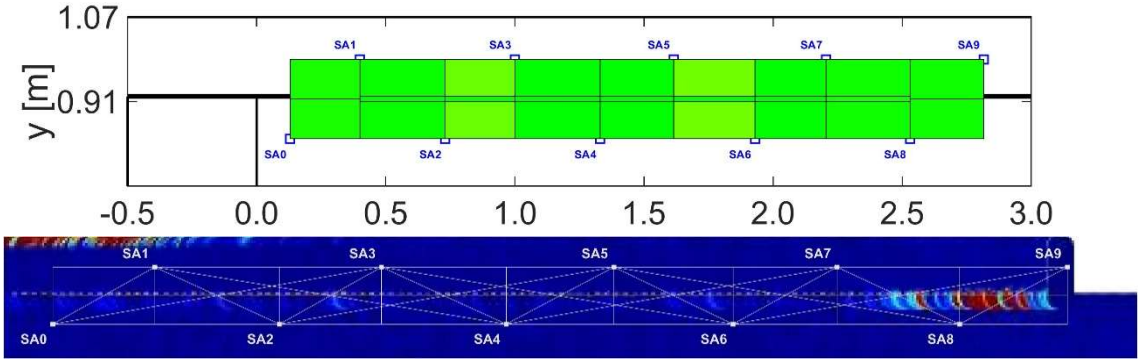




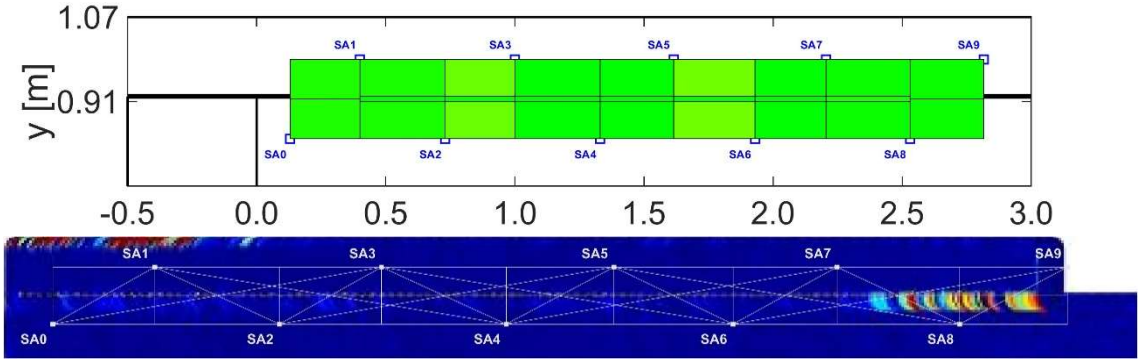
(d)- 100[kN]



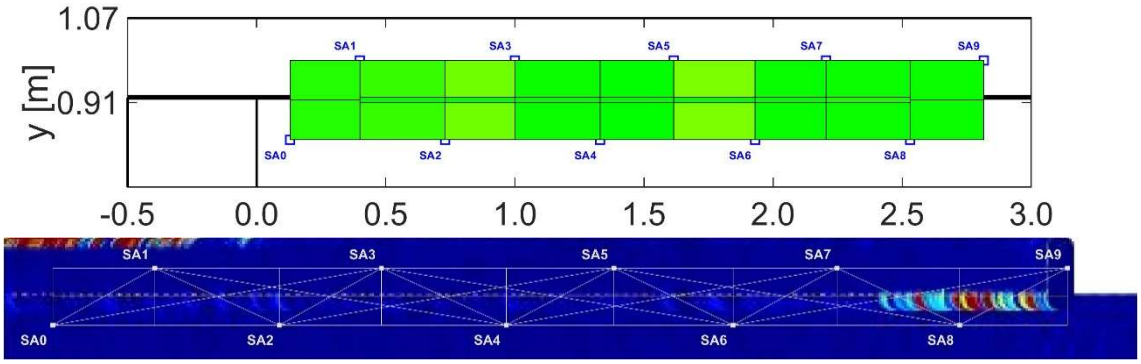
(e)- 150[kN]



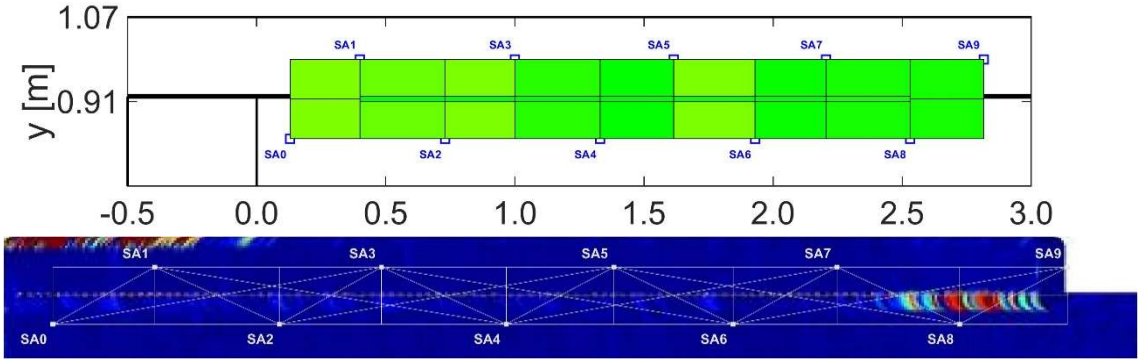
(f)- 200[kN]



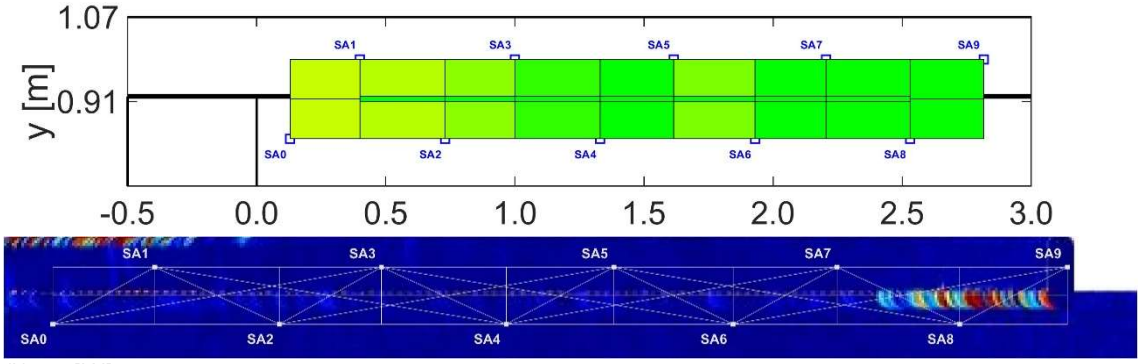
(g)- 250[kN]



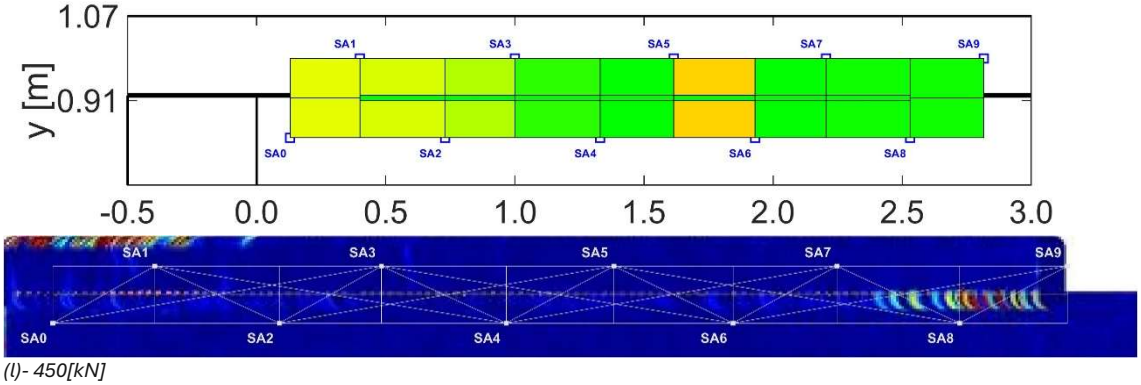
(h)- 300[kN]



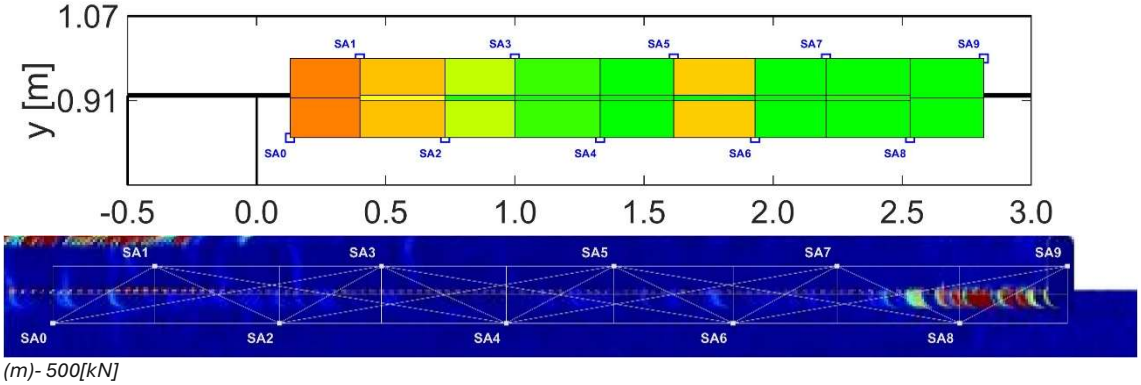
(i)- 350[kN]



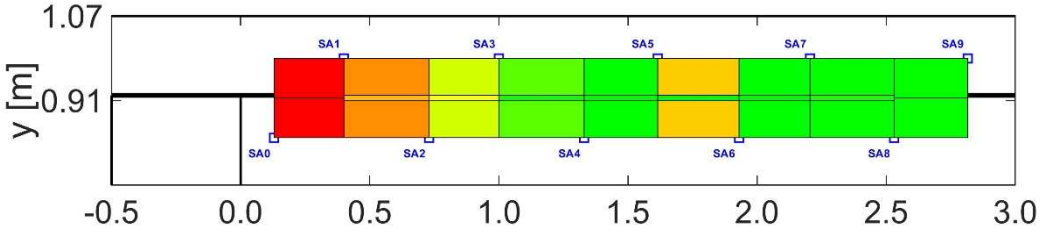
(k)- 400[kN]

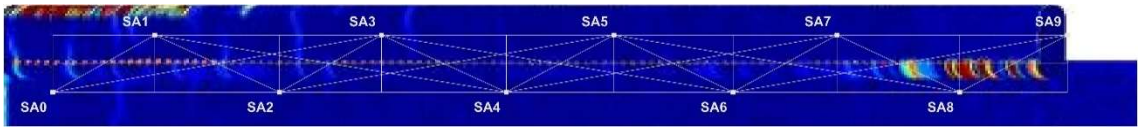


(l)- 450[kN]

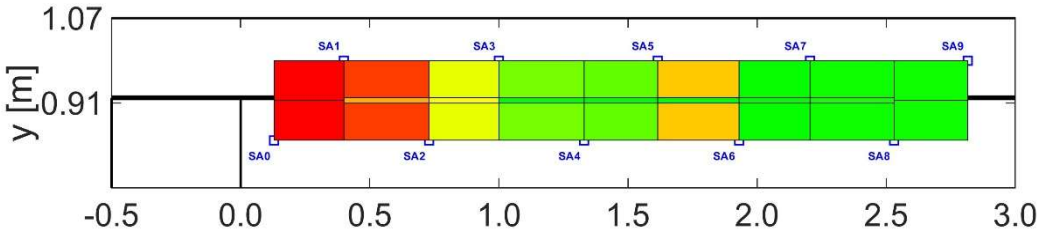


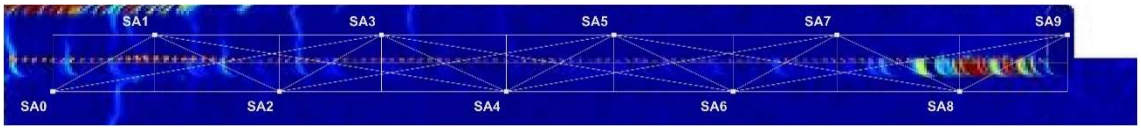
(m)- 500[kN]



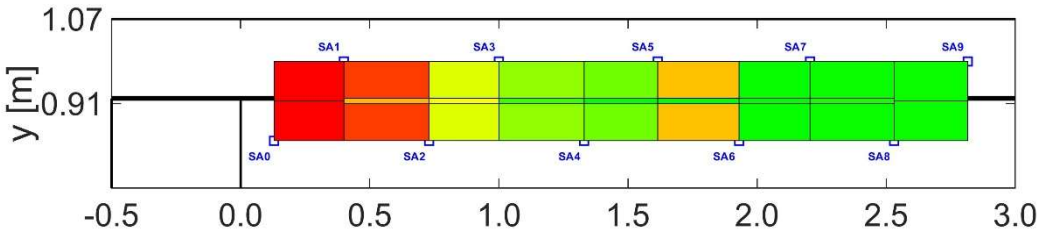


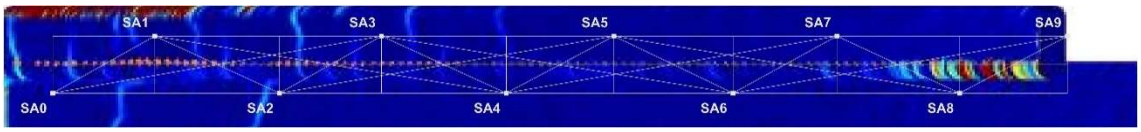
(n)- 550[kN]



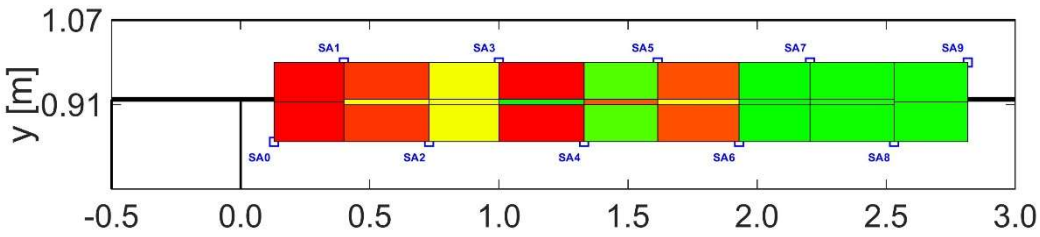


(o)- 600[kN]



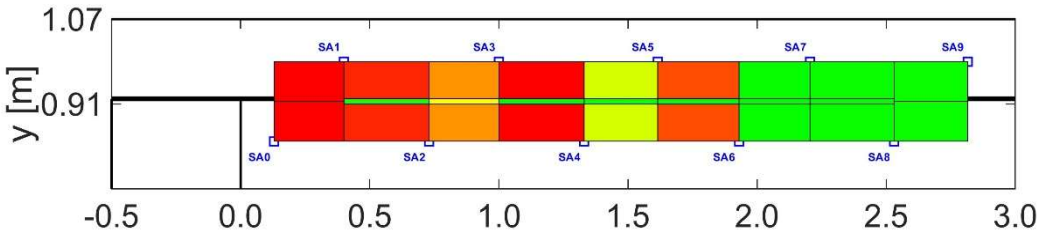


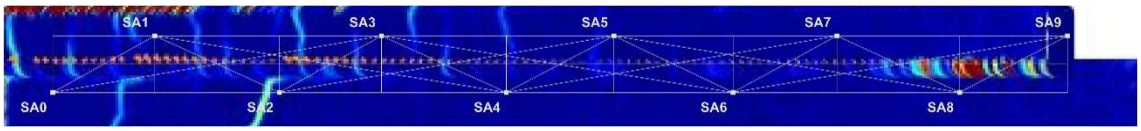
(p)- 650[kN]



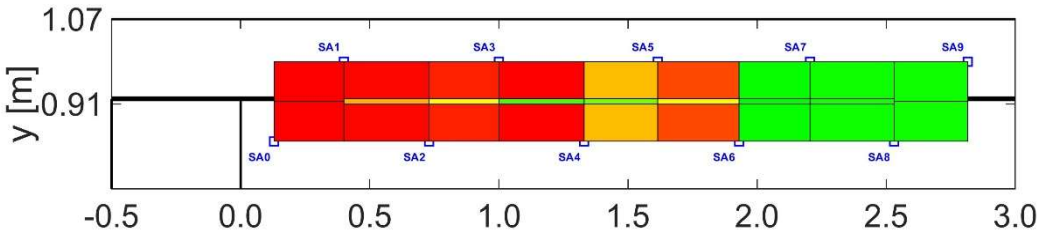
(no DIC Figure)

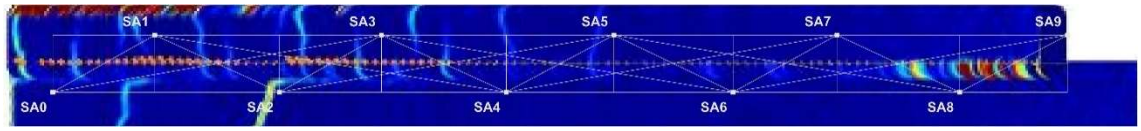
(q)- 700[kN]



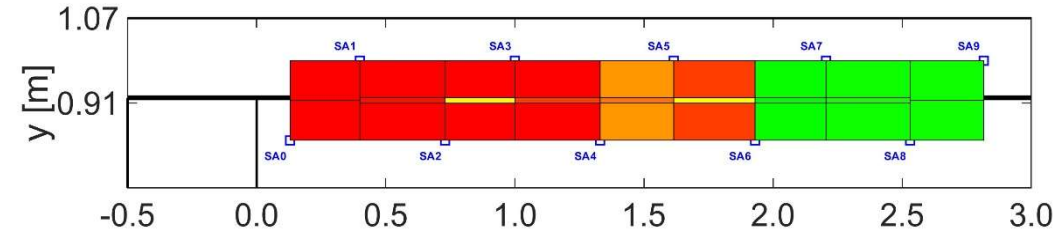


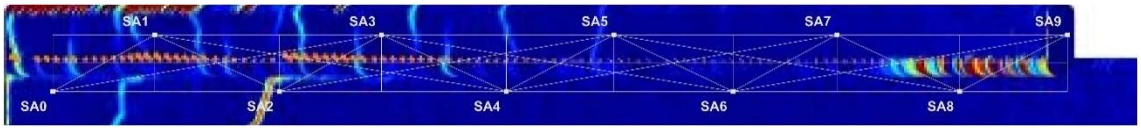
(r)- 750[kN]



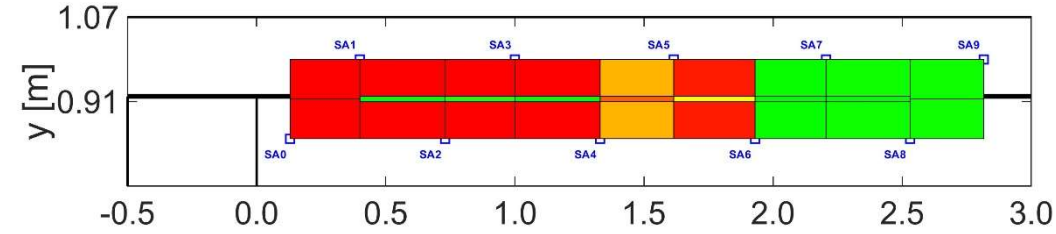


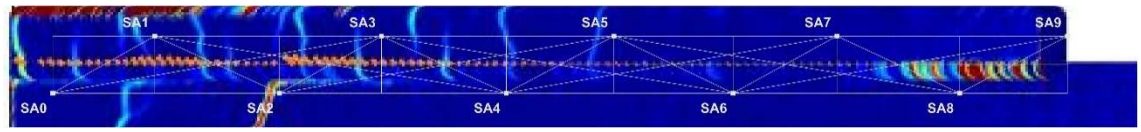
(s)- 800[kN]



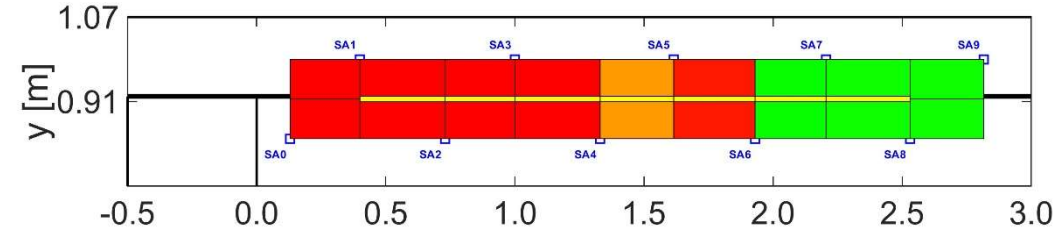


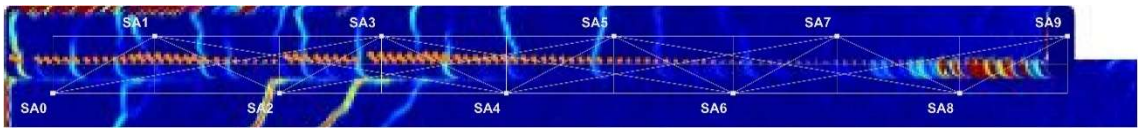
(t)- 850[kN]



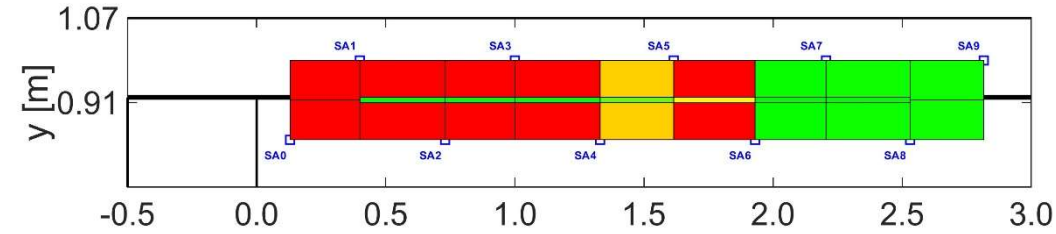


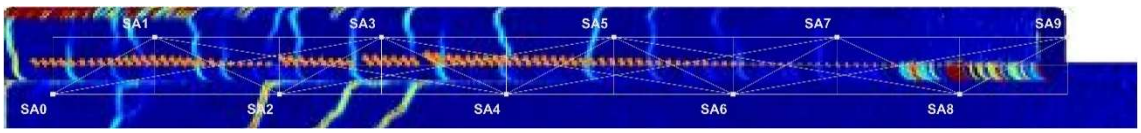
(u)- 900[kN]



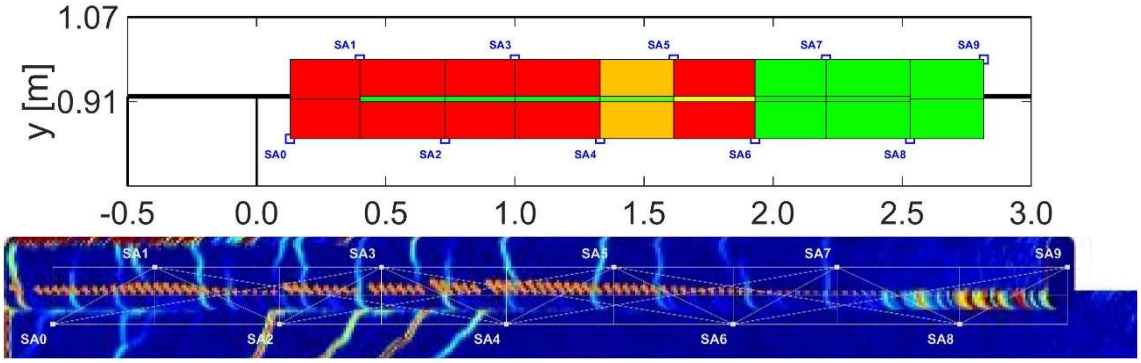


(v)- 950[kN]

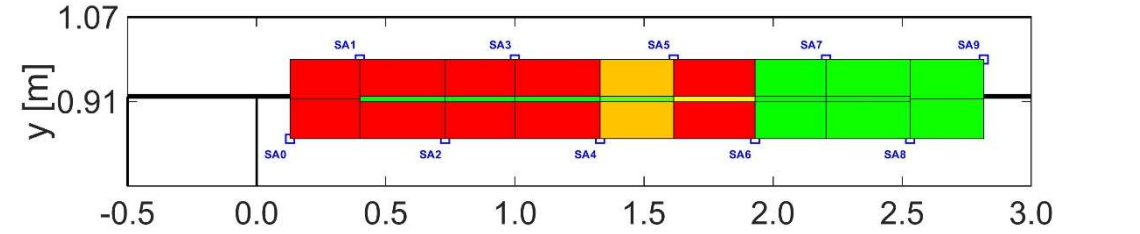




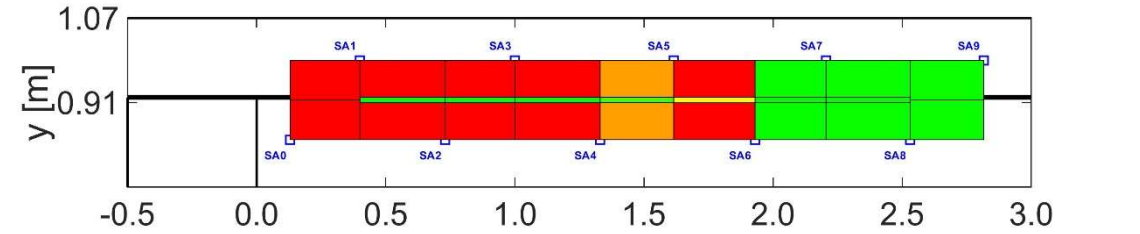
(w)- 1000[kN]



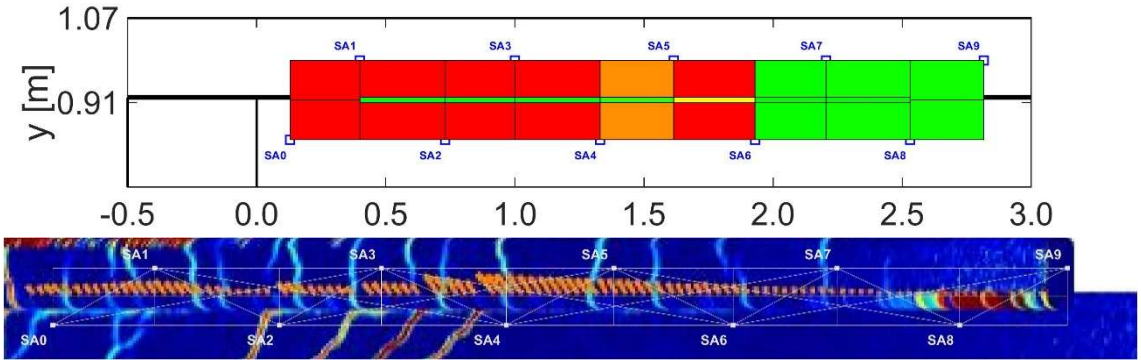
(x)- 1050[kN]



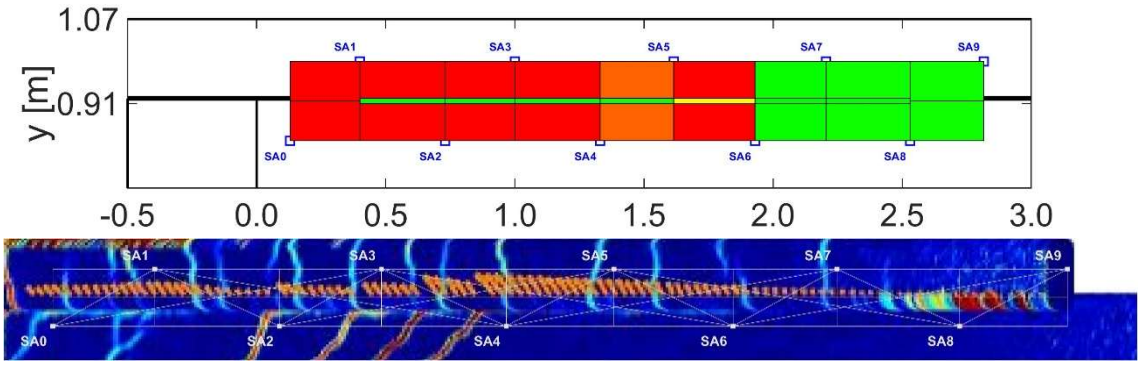
(no DIC Figure)
(y)- 1100[kN]



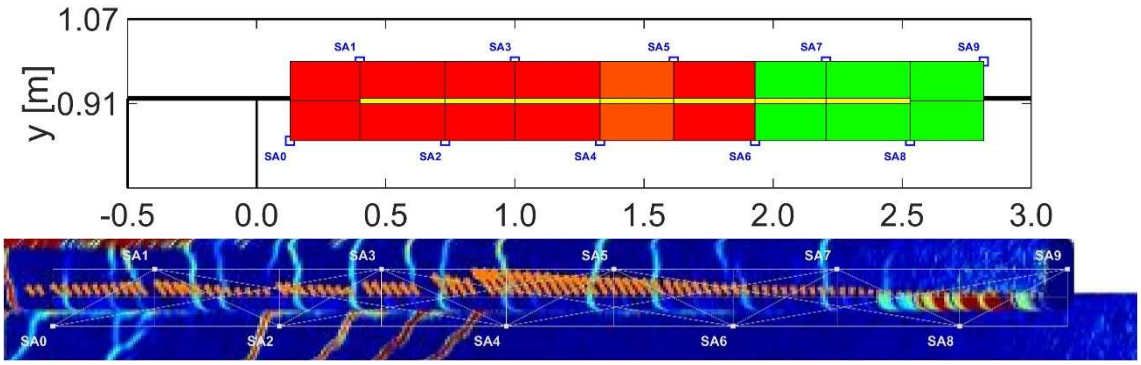
(no DIC Figure)
(z)- 1150[kN]



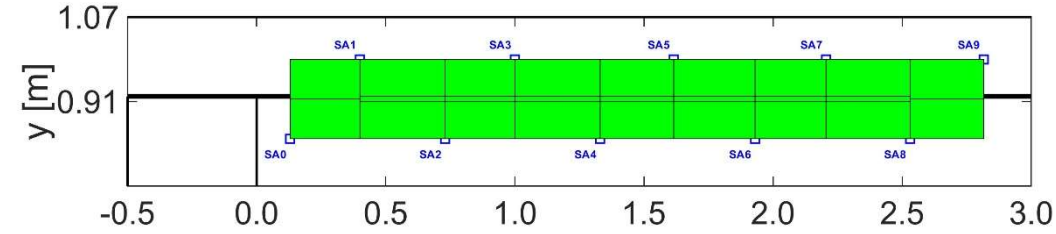
(aa)- 1200[kN]



(ab)- 1250[kN]



(ac)- 1300[kN]



(no DIC Figure)
(ad)-0[kN]