Institut für
Kontinuumsmechanik

High Performance Computing for Modelling of Stereolithography Process

Sandeep Kumar

Leibniz
Universität
Hannover

High Performance Computing for Modelling of Stereolithography Process

Sandeep Kumar

Leibniz
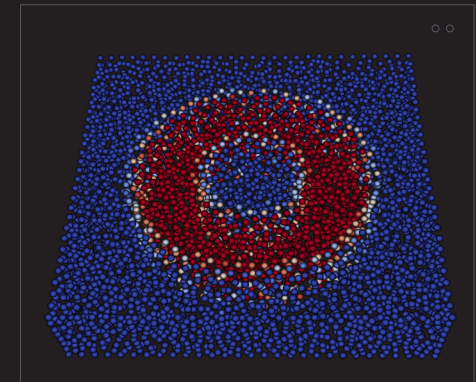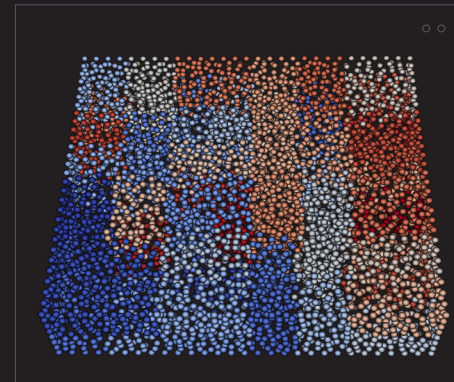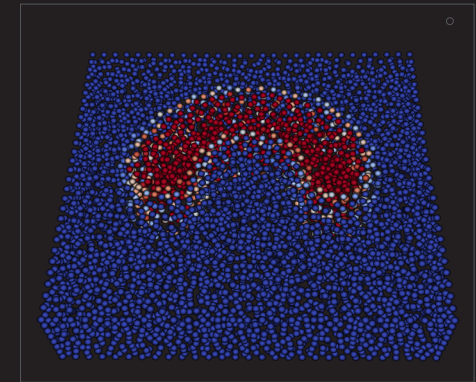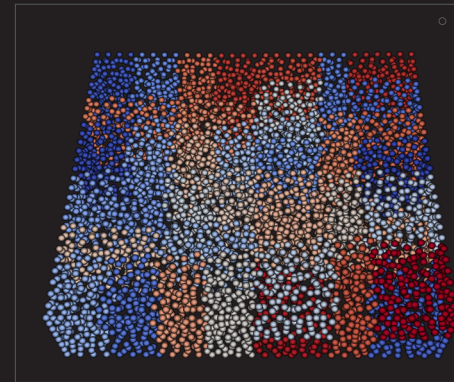Universität
Hannover

Institut für
Kontinuumsmechanik

In this multi-disciplinary work, the goal is to solve a computation intensive multiphysics problem using High Performance Computing (HPC) technologies while understanding the physics behind the Stereolithography (SLA) process simultaneously. A state-of-the-art 3D computational model based on a continuum approach is developed to investigate the SLA process in a multiphysics framework using Optimal Transportation Meshfree (OTM) method. This work studies the thermo-chemo-mechanical coupled evolution of the properties in the 3D printed product. This complex simulation model is developed in a HPC framework to utilize the potential of higher computing power. Parallel algorithms are implemented using Message Passing Interface (MPI). This work involves three new concepts to reduce the computational efforts: Efficient parallel implementation strategy of OTM with localized updates, use of flexible data structures for halo regions to handle variable workloads and reduced communication cost by use of neighbor collectives.
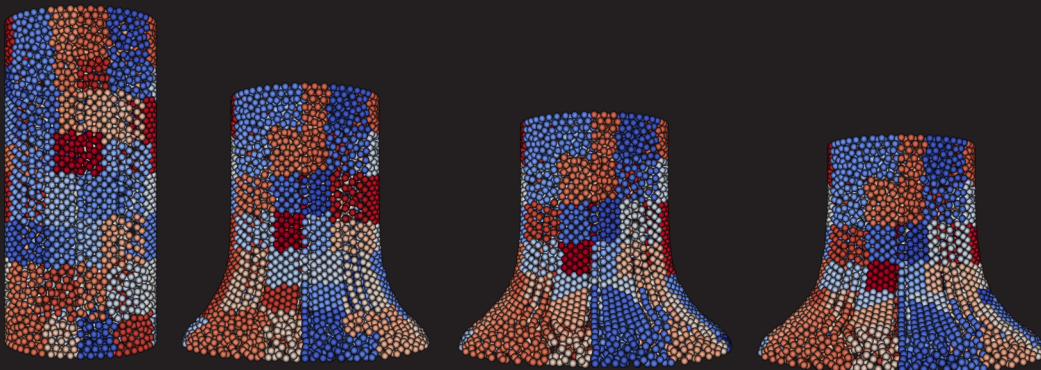
Sandeep Kumar | High Performance Computing for Modelling of Stereolithography Process

High Performance Computing for Modelling of Stereolithography Process

Von der Fakultät für Maschinenbau
der Gottfried Wilhelm Leibniz Universität Hannover

zur Erlangung des akademischen Grades
Doktor-Ingenieur

genehmigte Dissertation
von

M. Sc. Sandeep Kumar

geboren am 31.03.1991 in Rourkela, India

2022

To my parents

Zusammenfassung

In dieser Dissertation wurde ein hochmodernes 3D-Rechenmodell für den Stereolithografieprozess entwickelt, um die Entwicklung der Eigenschaften in einem Multiphysik-Rahmen mit der Stabilized Optimal Transportation Meshfree (OTM)-Methode auf der Grundlage eines Kontinuumsansatzes zu untersuchen. Um die Berechnungsleistung zu beschleunigen, wurde ein HPC-Rahmen für die OTM-Methode entwickelt. Der Stereolithographieprozess ist ein komplexer Prozess, da mehrere physikalische Prozesse daran beteiligt sind. In dieser Arbeit sind einige der Schlüsselphänomene, die in den Modellierungsrahmen einbezogen werden, die stark gekoppelte thermo-chemo-mechanische Entwicklung der Harzeigenschaften und die Ausbreitung des UV-Lasers durch das Harz. Die Photopolymerisation wird durch die Wechselwirkung des flüssigen Harzes mit dem UV-Licht angetrieben und erzeugt aufgrund ihrer exothermen Natur Wärme, die zum Aufbau mechanischer Spannungen führt. Die numerische und geometrische Komplexität, die sich aus diesen Phänomenen ergibt, stellt die gitterbasierten Techniken wie die Finite-Elemente-Methode (FE) vor große Herausforderungen und Komplikationen. Im Allgemeinen werden solche Probleme als Netzverzerrung bezeichnet. OTM-basierte Berechnungsmodelle sind die Lösung für diese Probleme. Die Methode ist recht neu auf dem Gebiet der Stereolithographie-Simulation und sie ist effizient bei der Erfassung der während des Druckvorgangs entstehenden Verformungen. Darüber hinaus reduziert die Parallelisierung mittels MPI mit dem Ziel der Skalierbarkeit auf großen CPU-Clustern den Berechnungsaufwand. Und die erzielten Ergebnisse führen zu hoch skalierbaren Ergebnissen. Das entwickelte Werkzeug kann zur Optimierung der Material- und Prozessparameter während des Druckprozesses eingesetzt werden, um eine verbesserte Genauigkeit der gedruckten Teile zu erreichen.

Schlagworte: High Performance Computing, Stereolithographie, Multiphysikalische Kopplung, Meshfree Methode

Abstract

In this dissertation, a state-of-the-art 3D computational model has been developed for Stereolithography process to investigate the evolution of properties in a multi-physics framework using Stabilized Optimal Transportation Meshfree (OTM) method based on a continuum approach. In order to accelerate the computational performance, HPC framework of the OTM method has been developed. Stereolithography process is a complex process in the sense that several physical processes are involved therein. In this work, some of the key phenomena incorporated in the modeling framework are highly coupled thermo-chemo-mechanical evolution of resin properties and propagation of the UV laser through the resin. The photopolymerization is driven by the interaction of fluid resin with the UV light and consequently generates heat due to its exothermic nature and resulting in building up of mechanical stresses. The numerical and geometrical complexities arising from these phenomena pose serious challenges and complications in grid-based techniques such as Finite element (FE). Generally, such issues are referred to as mesh distortion. OTM based computational modeling is one solution to these issues. The method is quite new in the field of Stereolithography simulation and it is efficient in capturing the deformations generated during printing process. Moreover, parallelization using MPI with an objective for scalability on large scale CPU clusters reduces the computational efforts. And, the obtained results leads to highly scalable results. The developed tool can be employed to optimize the material and process parameters during the printing process to achieve improved accuracy in the printed parts.

Keywords: High Performance Computing, Stereolithography, Multiphysical coupling, Meshfree Method

Acknowledgements

Hannover, March 2022   Sandeep Kumar

Contents

Chapter 1

Introduction

1.1 Motivation

In recent years, Additive Manufacturing (AM) technology has paved the way for digital manufacturing and since then, it has been the fastest-growing advanced manufacturing technologies in the world, (GIBSON ET AL., 2015). With the introduction of this technology, rapid manufacturing of complex and lightweight structures has become a reality. Compared to the traditional cutting and casting processes, this technology has significantly reduced the manufacturing costs, leading its way for applications in aerospace and other industries. AM technology make use of additive processes and combining materials layer-by-layer. This technology has revolutionized the product development by allowing to include intricate design features in the product. There are several AM technologies available, such as, Selective Laser Sintering (SLS), Electron Beam Melting (EBM) and Stereolithography (SLA). Among various AM processes, SLA is one of the most popular technologies nowadays, in which selective curing of the polymer resin using UV laser fabricates a 3D product. Two main advantages of SLA are its part accuracy and high surface finish. One can achieve higher control over the thickness of each layer. Broadly, there are two categories of studies conducted on SLA printed materials. In the first category, the study of process parameters and its influence on the mechanical properties of the printed part is carried out. Second study involves determination of mechanical properties of the printed part through development of empirical models.

The research on SLA has been motivated by predicting their development in order to achieve control over the mechanical properties of the product. Some of the challenges in the produced part include residual stresses, defect formation etc, which originate from several uncertainties, such as, UV intensity, resin composition etc, see (HUANG ET AL., 2015) for more details. However, there are limited simulation models to study the evolution of mechanical and chemical properties and to predict the accuracy of the cured part. A proper simulation model to study the thermo-chemo-mechanical coupled evolution of the properties of the cured product during photopolymerization has been proposed in this work. This work discusses the first ever developed 3D simulation model using meshfree tool to study the evolution of curing and mechanical properties through a viscoelastic material model and the key characteristics of the SLA process have been captured.

For execution of computational intensive numerical simulations, parallel architectures are important tools. In this work, this complex simulation model has been developed in a HPC Framework to utilize the potential of higher computing power. Also, the developed code allows to deal with larger model, thanks to the distributed memory usage. In order to reduce the computational time and to make full utilization of the available computing resources, HPC architecture has been adopted. Within HPC framework, parallel computing is performed by simultaneously executing computational tasks on multi-processor systems, (BOSSHARD ET AL., 2011). In this multi-disciplinary research, the goal is to solve this computation-intensive multiphysics problem using high performance computing technologies, while understanding the physics behind the SLA process simultaneously.

1.2 Additive Manufacturing

The term additive manufacturing, or AM, describes the use of additive processes, combining the material layer-by-layer. Also known as Rapid Prototyping which is due to the fact that whole product development process speeds up and not just in terms of time taken to build parts. The speeding up is also due to the fact that computers are being used throughout the development process. A 3D CAD modeling software is used as a starting point that describes the complete geometry of the part to be built. Also, reverse engineering techniques through laser or optical scanning can also be used for modeling. Usually, the CAD model is stored in surface tessellation language (stl) format, which describes the external closed surfaces of the model and forms the basis for calculation of the slices, see (GIBSON ET AL., 2015) for more details.

Various AM technologies have been developed over the time considering the type of the build material, such as, powder, solid sheet, molten metal, vat of liquid photopolymer, or inkjet deposited photopolymer. The use of vat photopolymerization is being widely used in several areas since its development in late 1960s. Photopolymer-based systems are quite easy to setup since it uses support made of the same photopolymer material that is used for the printed part. An advantage of these systems is that accuracy control of layer thickness is very good and very thin layers can be printed compared to other AM technologies. This, in turn, leads to high surface finish and fine precision. For the curing of commercial photopolymers, various types of radiations may be used, such as, gamma rays, X-rays, electron beams, UV and in some cases visible light. Most photopolymers react to UV range of wavelengths upon irradiation, thereby undergoing chemical reaction to become solid, known as photopolymerization. A detailed description follows in Chapter 2.

1.3 Meshfree Methods

1.3.1 Eulerian and Lagrangian Perspective

Mathematical models of a physical problem can be described in Eulerian and Lagrangian forms. In the Eulerian approach, the mesh or grid is fixed in space but in the Lagrangian approach, the mesh or grid is attached to the material which moves along with it. Handling

of moving or geometrically complex boundaries are treated easily using the Lagrangian approach, while it s a big challenge in the Eulerian approach. But, when it comes to physical problem of large deformations, Eulerian approach does not suffer from mesh distortions. On other hand, within Lagrangian approach, very large deformations lead to mesh distortions. In Lagrangian approach, material points carry themselves the field variables, hence, the time history of a variable associated to a material point can be tracked in a straight forward manner. Whereas, in the Eulerian approach, the variables are attached to a point in a spatial domain and not to a material point, for more details, see (LIU & LIU, 2003). Considering the characteristics of these two approaches, Lagrangian methods are preferred in modeling of solids and Eulerian methods are favored for computational fluid dynamics (CFD). Also, through another method known as arbitrary Lagrangian Eulerian (ALE) method (HUGHES ET AL., 1981), the advantages of both approaches are exploited in a combined way for problems involving fluid-structure interactions (FSI).

1.3.2 Meshless and mesh-based methods

In order to minimize the distortions in mesh-based methods resulting from large deformations, efforts were made to develop a new family of numerical methods with an aim to remove the mesh dependency entirely. These are known as meshless methods. In meshfree methods, a set of arbitrarily distributed points which are often called particles, is used to represent the body. The extensive quantities like stress, density etc. are not related to a mesh but stored at these discrete points (particles). More information about different meshless methods can be found in (WEISSENFELS, 2021).

1.4 High Performance Computing (HPC)

Realistic simulations of many engineering applications require large-scale computations. Computation on HPC clusters needs efficient and scalable codes. In order to utilize the full potential of the computing power of multi-core architectures, it is necessary to exploit both the intra and inter-node parallelism. Different parallel programming models were developed over the years. An overview can be found in (PRIMS ET AL., 2019). Currently, existing parallel programming models are based on distributed memory and shared memory platforms. Message Passing Interface (MPI) is most widely used standard paradigm on distributed memory platforms but it can also be applied to shared memory nodes. Other parallelization approaches exists, such as, OpenMP (for shared memory platforms), CUDA and OpenCL (for graphics processing units (GPUs)). Coupling our approach with OpenMP is possible and this will be the subject of future work.

1.5 Background and State of the Art

The description of the solidifying polymer has been captured by an advanced polymerization model by incorporating the viscoelastic behavior. The complexity of the SLA process

originates from its multiphysical nature, (WESTBEEK ET AL., 2018, 2020). The photopolymerization process starts with the exposure of UV light onto the liquid resin and the absorption of light limits its penetration depth into the resin. The liquid monomer polymerizes at those locations where enough photons are available to be absorbed by the photoinitiator. This process is accompanied by an exothermic reaction leading to temperature increase and chemical shrinkage, see (NARAHARA ET AL., 1999). Hence, capturing the combination of these complex multiphysical processes with simulation tools helps to predict the quality of the printed part. Several authors have worked on the modeling of the printed part for SLA process. The starting works include the development of cure depth model by Jacob ((JACOBS & REID, 1992)). In subsequent works, layered effects of the SLA process have been studied, see (CHAMBERS ET AL., 1995; HUANG ET AL., 2004; JIANG ET AL., 2006). Some works have incorporated the physics to predict the polymerization profile as a function of the UV laser source, see (JARIWALA ET AL., 2011; KANG ET AL., 2012, 2004).

Most of studies for the SLA process is focused on experimental investigations of the effects of material and process parameters. Also, the applications of numerical modeling approaches to understand the SLA process is not new. Several authors have worked on broad range of techniques, both empirical and numerical, starting from Jacobs' cure depth model (JACOBS & REID, 1992). They use a simplified model where it is assumed that the extent of cure is a function of the amount of exposure to UV radiation and the coupled transient, thermal and chemical effects are neglected. Existing research can be distinguished by two different approaches: First, attempts were made to capture the layering effect of the SLA process (using Finite Element Method) while considering all the layers to be homogeneous, for more details, see the works by (TARABEUX ET AL., 2018; YEBI & AYALEW, 2015). In the second approach, attempts were made to incorporate more detailed physics to study the spatial evolution of the degree of cure as a function of the UV laser source, see (JARIWALA ET AL., 2011) for more details. The general approaches for the cure models only contribute towards the evolution of thermal properties and are not coupled to the evolution of the mechanical properties of the printed part. Additionally, these simulations are performed on a finite element mesh, whose geometry is known in advance. Hence, the predicted printed geometry is only a deformed version of the initial geometry, where the processes, such as, illumination profile and light absorption can induce significant effects on the mesh geometry.

Due to the complexity and large scale of SLA simulations, large scale parallel computing using a message passing programming model has to be employed. Existing work on parallel architecture has been performed for fluid flow and heat transfer problems. Several parallelization strategies exist for CFD codes based on MPI and studies on their parallel performance have been done. In the work by (WANG ET AL., 2008), development of subdomain boundary mapping procedure and data communication procedures for multi-block structured grids have been discussed. Another work for flood flow simulations with hybrid MPI/OpenMP framework in Finite Element has been developed, for more details, see (SHANG, 2011). These hybrid models of MPI/OpenMP is aimed for coarse and fine-grained parallelism. The simulation of the granular media, made of discrete particles, was studied by (MAKNICKAS ET AL., 2006). Here, spatial domain decomposition strategy and data communication for particles in Discrete Element Method (DEM) have been implemented and the parallel performance has been discussed in detail. Another such parallel software

DEMFLOW has been developed for Discrete Element Method which studies the parallel performance considering the aspects of its load-balancing and particle-data reorganization strategies, see (WRIGGERS & AVCI, 2020) for further details. MPI libraries have large set of runtime parameters which need to be optimized for better performance. A parameter which has a large impact on the parallel performance is the amount of internal buffer space allocated for communication, see (JIN ET AL., 2011) for more details. The data size per process decreases with the increase of MPI processes resulting in reduced memory usage but this trend is observed only upto a certain point. With further increase of MPI processes, MPI buffers for data communication increases, leading to increased memory usage. A recent development of hybrid MPI/OpenMP framework for Optimal Transportation Mehsfree (OTM) method has been done by (LI ET AL., 2014).

The aim of this work is to develop and present a unified computational approach for the SLA process modeling based on the Optimal Transportation Meshfree (OTM) method. The distinctive characteristic of this research work is that it predicts the printed geometry on a full scale along with the generated deformations. This is achieved through multi-physical modeling of UV irradiation, curing and shrinkage. Apart from development of a computational model for the SLA process, parallel algorithms are implemented using Message Passing Interface (MPI). The Recursive Coordinate Bisection (RCB) algorithm is utilized for domain decomposition and for implementing dynamic load-balancing strategy. This work involves three new concepts to reduce the computational efforts: Dynamic halo regions, Efficient data management strategies for ease of addition and deletion of nodes and material points using advanced STL container, and nearest neighborhood communication for detection of neighbors and communication. Also, Linked Cell approach has been implemented to further reduce the computational efforts. Parallel performance analysis is investigated for challenging multiphysics applications like Taylor rod impact, serrated chip formation process and Stereolithography process. For all these applications, parallel implementation leads to high scalable results.

1.6 Structure of this work

This work contains five main more chapters as follows: In the second chapter, a detailed description of the physical phenomena behind the SLA process are explained. Subsequently, in the third chapter, considering the complexity of the multi-physical process, the included and excluded modeling objectives are mentioned. Also, the classical continuum framework for the material modeling is introduced, which includes the related kinematics and the conservation principles. Here, the modeling of the resin behavior including the classical models for the large strain viscoelasticity has been introduced. This includes chemical consideration, which is the basis for the photopolymerization modeling and defining a thermo-chemical free energy function for the thermo-chemical consideration. The fourth chapter starts with a brief introduction to the OTM method. In the remainder of this chapter, the formulation of the stabilized optimal transportation meshfree method has been discussed. In the fifth chapter, the development of parallel framework for the OTM method with MPI has been discussed in detail. This includes the process of domain decomposition and communication,

object-oriented framework and special architecture considerations for the OTM method. In the later part of this chapter, the scalability performance of the parallel framework have been studied on two test examples. At the end, we conclude the dissertation in the sixth chapter with the simulation of the stereolithography process and the outcomes of this work. Besides, suggestions have been made for possible extensions for the future research.

Chapter 2

Stereolithography

Stereolithography (SLA) is one of the Rapid Prototyping (RP) technologies which are widely used in industry now. SLA uses a setup where a computer assisted UV laser source is used to induce curing of liquid photopolymer resin layer by layer (WANG ET AL., 2017; MELCHELS ET AL., 2010). The platform then moves downward alongwith the first cured layer. Subsequently, fresh liquid resin flows over the cured part and the process is repeated again to form a new cured layer over the previous layer. The factors affecting the quality and resolution of the printed part are intensity of laser power, duration of UV exposure and scan speed.

2.1 Laser Configurations

The stereolithography process employ two distinct methods of irradiation: vector by vector based scanning and mask projection based scanning approach, see (FOUASSIER, 1995; BHATTACHARYA, 2000; DA SILVA BARTOLO, 2001; BARTOLO & MITCHELL, 2003) for more details. The vector scanning approach illuminates a small portion of the resin with the help of a moving actuator. This approach has relatively high precision because the layer is built in a point-by-point style with a very fine laser beam spot. However, this scanning approach slows down the process. In mask projection based approach, the whole layer is formed in one exposure, ZHOU ET AL. (2015), through projection of two-dimensional array of beams. Figure 2.1 illustrates both types of approach. In current work, vector scanning approach has been used.

2.2 Laser-Resin Interaction

The UV curing process depends on laser intensity and exposure time. Distribution of UV laser exposure (laser intensity) is an essential condition to determine the cured areas and thereby the printed part dimensions. A working curve based on two parameters that govern the photopolymerization reaction has been laid out by Jacobs (JACOBS & REID, 1992): penetration depth of the curing light and the laser intensity required for photopolymerization. In this work, UV laser beam is considered as a Gaussian laser beam. According to the Gaussian law, its intensity decreases from the center of the beam. Figure 2.2 shows the Gaussian en-
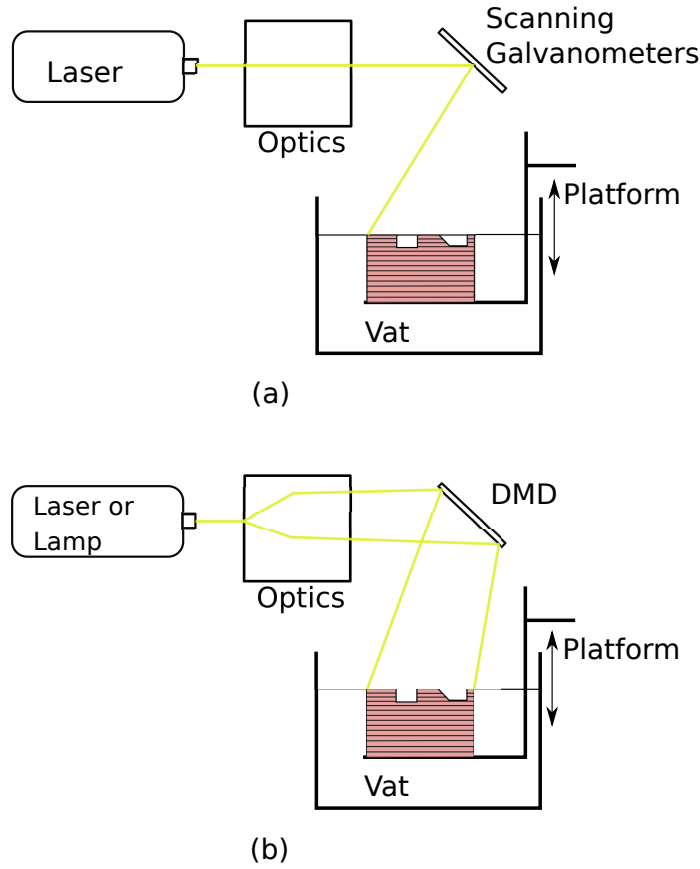
Figure 2.1. Two types of illumination approaches for stereolithography process: (a) Vector based scanning process and (b) mask projection based scanning process

ergy distribution E, which is equivalent to the intensity over time. Due to the radiation effects of UV laser exposure, the cured part can be described as a parabolic cylinder, see Figure 2.2, where C_d is the cure depth and C_w is the cure width, see (PHAM & GAULT, 1998; JACOBS & REID, 1992).

The UV laser energy follows a Gaussian distribution on the material's surface and is defined as the radiant laser energy per unit surface area (mJ/cm^2). The decrease in UV laser intensity with depth (resin thickness) due to photo-absorption is assumed to obey Beer-Lambert law as shown in the following equation, see (PERRY & YOUNG, 2005):

$$\mathrm{E}_{Laser} = \frac{2\mathrm{PT}}{\pi W_0^2} e^{\left(-\frac{2y^2}{W_0^2} - \frac{z}{\mathrm{D}_p}\right)} \tag{2.1}$$

where E_{Laser} is the UV laser energy at any point (y, z) on the material's surface, P is the UV laser power, T is the exposure time, W_0 is the radius of Gaussian laser beam and D_p represents the penetration depth coefficient which corresponds to the UV penetration depth into the material until the intensity of Gaussian laser beam is reduced by $\frac{1}{e}$, relative to the laser intensity on the material's surface, see (TARABEUX ET AL., 2018). The penetration depth depends on the absorbance characteristics of the resin and is related to the resin's

composition, see (BENNETT, 2017) for further details.

At the surface, $y = 0$ and $z = 0$, the UV energy is maximum

$$\text{E}_{max} = \text{E}(0,0) = \frac{2PT}{\pi W_0^2} \tag{2.2}$$

where, the peak light intensity is defined as $I_0 = \frac{2P}{\pi W_0^2}$. The maximum cure depth, also known as "working curve", can be derived at $y = 0$, (LEE ET AL., 2001)

$$\text{C}_d = \text{D}_p \ln\left(\frac{\text{E}_{max}}{\text{E}_c}\right) \tag{2.3}$$

where, E_c is the critical UV energy required to initiate the photopolymerization and C_d is the depth of the cured resin. It is seen from Equation (2.3) that cure depth depends on E_c, D_p and E_{max}. E_c and D_p are material specific properties and E_{max} is related to the input UV energy.
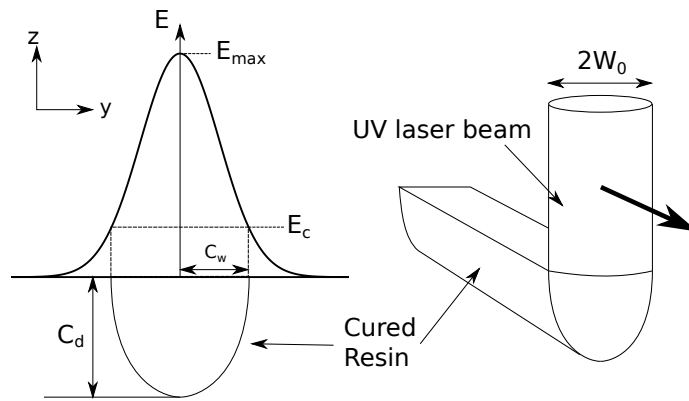
Figure 2.2. Illustration of Gaussian energy distribution

2.3 Photopolymerization Materials

Photocuring or photopolymerization is defined as a process of rapid synthesis of polymers by chain reactions upon the absorption of light by a polymerizable system, (SCHNABEL, 2007). Photopolymerization processes of thermosetting resins creates an insoluble, infusible, and highly crosslinked 3D network, (SELLI & BELLOBONO, 1993). Photocrosslinkable materials usually consists of muntlifunctional monomers or oligomers, with small amounts of UV sensitive photoinitiator in order to generate free radicals or ions upon UV exposure, (DECKER, 2001). Photoinitiators play the role of converting the physical energy of the incident UV light into chemical energy by forming reactive intermediates, see (GIBSON ET AL., 2015). These materials are divided into two groups based on their polymerization mechanism: (1) materials which cure by free-radical mechanism and (2) materials which cure by ionic mechanism. In free-radical mechanism (Figure 2.3), photoinitiated radical polymerization of monomers (such as, acrylates and unsaturated polyesters) takes place. Acrylate based UV-curable resins are most widely used, which is due to their high reactivity, resulting

in short reaction times in order of fractions of a second, (MENDES-FELIPE ET AL., 2019). In ionic curing mechanism, photoinitiated cationic polymerization of vinyl ethers and multifunctional polymers takes place.
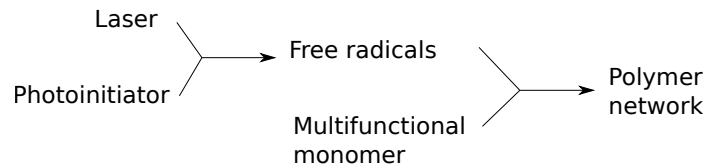
Figure 2.3. Basic principle of laser-induced free-radical mechanism

There are several methods to activate a free-radical photopolymerization process, such as, by light, chemical redox triggers, voltage and mechanical force. Among these methods, light is used as an illumination source to trigger the free radical polymerization in several engineering applications, such as 3D printing (CHEN ET AL., 2016).

2.4 High-speed UV Curing Process

In polymerization processes, ultraviolet light proves to be a powerful tool to initiate chemical reactions. Laser-induced polymerization allows fast transformation of a liquid resin containing UV sensitive photoinitiators and monomers into a cross-linked solid (DECKER, 2001). Photopolymerization by UV radiation is advantageous due to its accelerated processing time (rapid solidification), higher-energy efficiency and overall "environmental friendliness", and better controllability (YEBI & AYALEW, 2015; MENDES-FELIPE ET AL., 2019). Some of the other key features of laser-induced polymerization are its spatial and temporal control of the reaction. Due to the spatial coherence of the laser emission, the beam can have great directivity, which can be focused even to a tiny spot of micronic dimensions. Also, since the light-intensity does not change with distance, non-planar objects can be uniformly illuminated irrespective of the distance from the laser. Due to temporal coherence of the laser beam occurring at a precise wavelength, an accurate control of penetration depth can be achieved by unwanted side reactions, (DECKER, 1999).

During radical photopolymerization, polymerization process ceases immediately following the discontinuation of irradiation (KIM ET AL., 2020), which means very limited dark cure potential. This process of immediate termination gives rise to spatial and temporal control that is highly desirable in Stereolithography. The free radical process produces short propagating radical lifetimes due to termination reaction, see (KIM & STANSBURY, 2008). The rapid cessation of the movement of free radicals occurs immediately after the formation of polymeric chains. These are known as trapped radicals. Due to this phenomena, the polymerization reaction stops since the free radicals cannot interact further with monomers. Hence, the regions of the monomer which are exposed to UV light polymerize and solidify, while the rest remains in fluid state. As photopolymerization proceeds, the viscosity of the mixture in the direct vicinity of the UV light increases untill the point of vitrification and the material becomes solid (FLACH & CHARTOFF, 1995).

2.5 Kinetic Interpretation of UV Curing Reactions

Analytical models of the UV curing process are used to describe the coupled cure kinetics and heat transfer of thermosetting resins. When the resin formulation containing the UV sensitive photoinitiators and monomers are irradiated with UV, photopolymerization reaction occur. Excited radicals are formed due to the absorption of photons by photoinitiator molecules. The exothermic cure reaction is characterized by two main transitions, (SELLI & BELLOBONO, 1993):

- Gelation (liquid-to-rubber transition)

- Vitrification (liquid- or rubber-to-glass transition)

Gelation is a non-reversible process leading to the formation of an infinite molecular network and dramatic increase in viscosity. This result in existence of two phases: a sol phase and a gel phase. During the vitrification process, a glassy solid material is formed resulting from increase in cross-linking density and the molecular weight of the polymer being cured (DA SILVA BARTOLO, 2007). Subsequently, the rate of reaction is diffusion controlled and therefore, curing in the glassy state drastically slows down, see (LAPIQUE & REDFORD, 2002). Due to vitrification, the reaction shifts from kinetically controlled to diffusion controlled (PRIME, 1997).

In order to measure the curing reaction in radical polymerization, the degree of cure (α) is introduced.

$$\alpha(t) = \frac{H(t)}{H(\infty)} \,\epsilon\, [0, 1] \tag{2.4}$$

where, $H(t)$ is the accumulated released heat at time t and $H(\infty)$ is the total accumulated released heat of the cured material. The degree of cure also represents the mass fraction of the already cured material.

Usually, the cure kinetics can be modeled through three different approaches:

- Energetic models,

- Mechanistic and semimechanistic models, and

- Phenomological models.

Energetic models The energetic models are based on the fact that the initiation of the curing process occurs only when a critical value of energy is achieved. The factors considered in these models are based on relation between radiation profile, radiation intensity, and energy, that could simulate both mask irradiation and direct irradiation processes, for more details, see (MATIAS ET AL., 2009).

Mechanistic and semimechanistic models   Mechanistic or semimechanistic models are based on both the concept of free radical polymerization and mechanism of reactions with diffusion (MATIAS ET AL., 2009). The complexity of the curing reaction is simplified through several assumptions and approximations. Incorporating the effect of diffusion, free volume molecular parameters and the glass transition temperature are introduced, which are used to modify the rate constants of the free radical polymerization reactions, see (ZETTER-LUND & JOHNSON, 2002; DA SILVA BARTOLO, 2007). After each change of resin formu-lation, the kinetic parameters need to be recomputed since these models do not include the effects of initiator concentrations on the degree of cure.

Phenomological models   Phenomological models were developed to simulate the physical and chemical changes occurring in the material and its surroundings when exposed to UV radiation. In this model, law of conservation of energy has been described by coupling the heat transfer phenomena with kinetic models for the degree of cure. Several possibilities exist to formulate the kinetic models for degree of cure, see (KIASAT, 2000). A theoretical kinetic model of the cure reaction rate of epoxy acrylates can be interpreted as given by the dual Arrhenius approach, see (KAMAL & SOUROUR, 1973; SOUROUR & KAMAL, 1976). This model represents the autocatalytic cure kinetics

$$\dot{\alpha}(\Theta) = [A_1(\Theta) + A_2(\Theta)\alpha^m] (1 - \alpha)^n \tag{2.5}$$

The autocatalytic cure kinetics model in Equation (2.5) represents the maximum value of the cure reaction rate at an intermediate conversion ($\alpha > 0$), see (ZHAO & HU, 2010).

$$A_1(\Theta) = A_{c1}e^{-\frac{B_1}{\Theta}} \text{ and } A_2(\Theta) = A_{c2}e^{-\frac{B_2}{\Theta}} \tag{2.6}$$

Rate constants A_1 and A_2, can be related to Arrhenius temperature dependency, see Equation (2.6), where Θ is the reaction temperature and A_{c1}, B_1, A_{c2} and B_2 are material parameters influencing the cure reaction rate. The specific temperatures are defined as $B_i = \frac{\Delta E_i}{R}$, where ΔE_i are the activation energies and $R = 8.314459 \frac{J}{mol \cdot K}$ is the universal gas constant.

During the development of the numerical simulation model for UV-curing, the spatial atten-uation of UV intensity with resin depth due to photo-absorption has been neglected. Hence, there is no scattering phenomenon and the UV intensity is considered to be uniform across the resin depth, see Figure 6.1. Another consideration for the heat transfer model is that heat generation due to the direct absorption of UV radiation is introduced, in addition to exothermic cure reactions.

Chapter 3

Mathematical Framework and Governing Equations

3.1 Multi-physical coupling

In order to establish an appropriate model for the Stereolithography, we need to understand the physics of the process and the associated assumptions. There are several works discussing the mathematical/ physical modeling of UV curing process for Stereolithography, see (MATIAS ET AL., 2009; TANG, 2005). The physics involved in the Stereolithography process has been discussed in detail focusing on the illumination source and its effect on the polymerized profile, for more details, see (JARIWALA ET AL., 2011; KANG ET AL., 2012, 2004). Most of the formulations used for the material modeling of Stereolithography in this chapter is referred from the work by (HARTMANN, 2019). Starting from the gel point, the evolution of viscous, elastic and plastic properties can be described as a function of the degree of cure. Mechanical properties start to build up during the solidification of the resin. Combined influences of thermal expansion due to exothermic photopolymerization process and chemical shrinkage due to solidification results in stress build up.

The process of free-radical photopolymerization is a complex physical and chemical process, in which a liquid monomer solution is converted to a solid polymer rapidly. The mechanism starts with UV exposure of the photoinitiator in the solution and its decomposition into active (free) radicals, which in turn attach to the monomers to initiate the polymerization reaction. The chemical reaction is quite fast with a high degree of cure since the crosslinked network is reached within 20ms of UV exposure, (FLACH & CHARTOFF, 1995), which restricts the further movement of free radicals into the monomer solution, thereby limiting the curing process. Due to the growth of polymer chains, they start to connect and form a network structure which inturn results in a dense solid. During this transformation of the material, its properties changes dramatically.

So, we face a coupled multi-physics problem. However, when we take a closer look, this multi-physical process can be divided into four submodels by taking a modular approach:

- Irradiation model

- Photopolymerization model

- Temperature model

- Material model
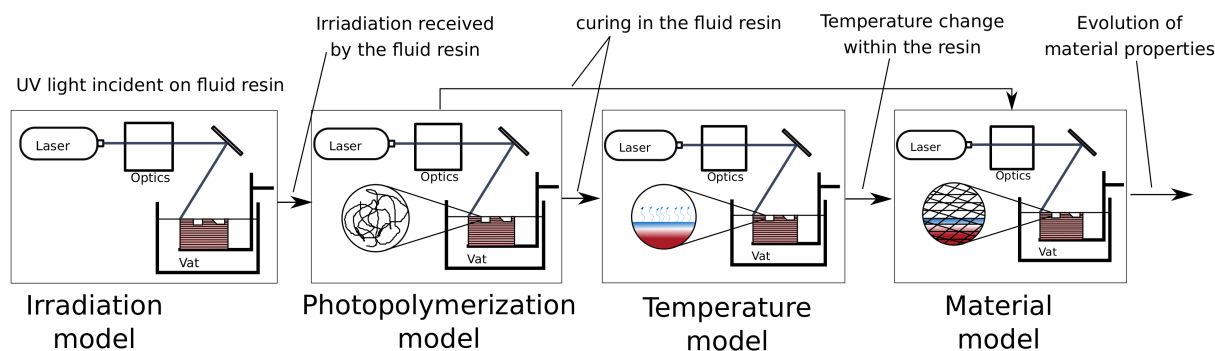
Figure 3.1. Schematic representation of photopolymerization model, consisting of four submodels with their respective input and outputs.

Irradiation model: In the irradiation model, the light intensity varies with changes in the spatial position in the solution and it can be described by Beer-Lambert Law, see Section 2.2 for more details. The UV irradiation is a surface phenomena and the light penetrating into the solution is absorbed by the photoinitiators, photoabsorbers and other light reactive species in the solution. During this process, light intensity is attenuated.

Photopolymerization model: In the photopolymerization model, the main principle is the light initiated polymerization of a liquid monomer into solid polymer (CLASSENS ET AL., 2021). Within fractions of a second, the liquid-to-solid phase transition of the liquid monomer solution takes place. In order to link the photopolymerization process with the evolution of material properties, suitable characterization parameters needs to be defined for the evolution of the polymerization system. Using the degree of cure of the monomers or the functional groups, several material properties can be linked as a function of the degree of cure. The degree of cure is used as an internal variable to quantify the amount of conversion of monomer into polymer and it is easier to measure than the polymer chain distribution. It is a good indication of the average composition of the system. For more details, see Section 2.5. The photopolymerization process is also controlled by the diffusion of the curing solution's species. As the viscosity of the system increases due to growth of polymer chains, the diffusion ability of free radicals into the monomers is restricted, which causes decrease in propagation rate of photopolymerization process.

Temperature model: Photopolymerization is an exothermic process. Factors which affect the rate of cure reaction are the local intensity of UV radiation and the duration of exposure over the target resin material. In addition, heat transfer phenomena is also involved, which includes heat conduction in the resin, heat generation by the exothermic cure reaction and convection heat transfer between the resin and the surroundings. Material specific thermal effects are also observed due to this exothermic process of the curing reaction. Generally

photopolymerization processes are insensitive to thermal changes and thermal decomposition of most photoinitiators does not produce enough free radicals to initiate polymerization process, see (PERRY & YOUNG, 2005). But, some photoinitiators are sensitive to high temperatures and in this case photoinitiator decomposition could occur thermally at high temperatures. In the temperature model, the temperature gradients due to exothermic cure reaction and resulting diffusion in the body are considered. The heat convection is a surface phenomena taking place when the surface temperature differs from that of the surroundings.

Material model: During the liquid-to-solid phase transition process, material properties such as, polymer chain length, molecular weight and crosslinking density, evolve with the degree of cure. These changes affect the macroscopic material properties such as relaxation time, Young's Modulus and volume shrinkage. In the material model, the photopolymerization process is linked with the material property evolution. During the polymerization process, new crosslinks are formed in addition to old crosslinked polymer network. Hence, this is a continuous microstructure evolution process coupled with mechanical deformation, such as curing shrinkage deformation. To describe the evolution of material properties, degree of cure of the functional groups or monomers is used as an internal variable. This is a good indication of the average composition of the system and it is easier to measure as compared to the distribution of polymer chains.

A thermo-chemo-mechanical model is proposed to describe the physical and chemical phenomena involved in the photopolymerization process. The key assumptions in the modeling of the Stereolithography process are as follows:

- In the cure kinetics model for UV-curing, the spatial attenuation of UV intensity with resin depth due to photo-absorption has been neglected. The UV intensity is considered to be uniform across the resin depth. The UV intensity is considered to be maximum along the entire resin depth, see Figure 6.1. The maximum UV intensity on the resin surface has been taken into account by Beer-Lambert law.

- There is no optical scattering phenomenon and the material flow due to convection and diffusion has also been neglected.

- Direct absorption of UV radiation is introduced, as defined by the Beer-Lambert law.

- Contributions to internal heat generation is from both direct absorption of UV radiation and heat of polymerization (due to exothermic cure reactions).

- Prior to the UV radiation exposure, the temperature levels of the SLA machine are considered to be at room temperature.

- No residual (dark) polymerization occurs, which mean polymerization process terminates immediately when the irradiation source is extinguished (KIM & STANSBURY, 2008; FLACH & CHARTOFF, 1995).

- In our work, we are considering photopolymerization reactions are insensitive to the temperature and the reaction is solely dependent on photochemical decomposition.

- The curing process is independent of the depletion of photoinitiator concentration. Also, the presence of oxygen is inhibited which can ensure a good quality product during the simulation when compared to the ideal samples or actual printed samples.

3.2 Continuum Mechanics

Continuum mechanics is concerned with description of the behavior of solid body during deformation process under the influence of the environment (or surroundings). Continuum mechanics can be divided into three sub-domains. Pure geometric description of movement of the bodies can be described through kinematics. Second, the material body is treated as a continuous medium, where the interaction between the body and its environment is introduced through equilibrium equations (balance of linear and angular and linear momentum). Finally, a functional relationship between the kinematic quantities (such as strain) and the kinetic quantities (for instance, stress) is established through constitutive equations (material properties).

3.2.1 Kinematics

Kinematics describes the deformations (and motions) of the body over time in stress-free initial and current configurations through one-to-one mapping as shown in Figure 3.2. The initial configuration refers to the undeformed initial body B at time $t = t_0$ and it is defined as a set of material points P which occupies a connected region in a three-dimensional Euclidean space $\mathbb{R}^3$. Due to a process of deformation, each material point P is assigned to a position vector uniquely identified at each time t. For the current configuration of the body at time $t > t_0$, the material point P can be described by the current position vector $\boldsymbol{x}$. The deformation mapping between the initial and the current configuration is represented by bijective unique mapping $\boldsymbol{\varphi}$

$$\boldsymbol{x} = \boldsymbol{\varphi}(\boldsymbol{X}, t) \tag{3.1}$$

$$\boldsymbol{X} = \boldsymbol{\varphi}^{-1}(\boldsymbol{x}, t) \tag{3.2}$$

where $\boldsymbol{X}$ represents the position of each material point P in the initial configuration. And, the vector field $\boldsymbol{\varphi}$ is defined as the motion of the body, see (HOLZAPFEL, 2002), which can be seen as one-to-one mapping of the position vector between the initial and the current configurations of the body.

The displacement vector $\boldsymbol{u}$ of each material point P is determined by the difference of position vectors in the current and the initial configuration.

$$\boldsymbol{u} = \boldsymbol{x}(\boldsymbol{X}, t) - \boldsymbol{X}. \tag{3.3}$$

The velocity field and the acceleration field can be computed from the time derivatives of the displacement vector

$$\boldsymbol{v} = \frac{d\boldsymbol{u}}{dt} = \frac{d\boldsymbol{x}(\boldsymbol{X}, t)}{dt}, \ \boldsymbol{a} = \frac{d\boldsymbol{v}}{dt} = \frac{d^2\boldsymbol{x}(\boldsymbol{X}, t)}{dt^2}. \tag{3.4}$$
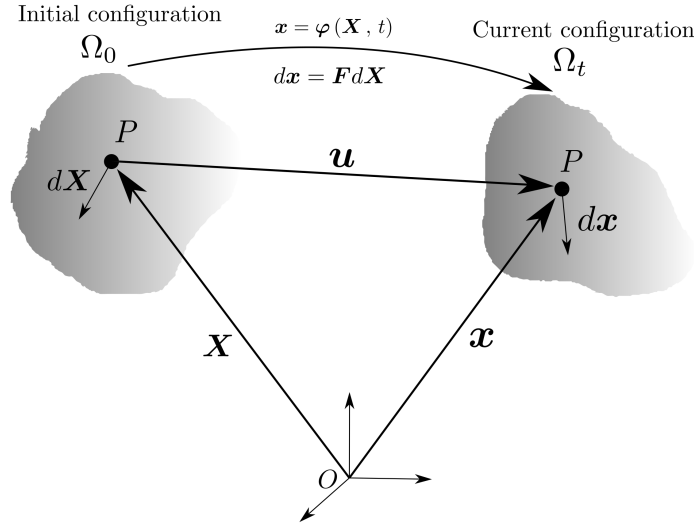
Figure 3.2. Schematic representation of deformation mapping φ defined w.r.t an initial configuration Ω_0.

In continuum mechanics, to describe the deformation process locally, the deformation gradient $\boldsymbol{F}$ is introduced, which maps infinitesimal vector element $d\boldsymbol{X}$ in the initial configuration to the corresponding vector element $d\boldsymbol{x}$ in the current configuration

$$\boldsymbol{F} = \frac{d\boldsymbol{x}\left(\boldsymbol{X}, t\right)}{d\boldsymbol{X}}. \tag{3.5}$$

where $\boldsymbol{F}$ is a two-point tensor since points in two configurations are involved. In order to ensure the non-singularity of $\boldsymbol{F}$, the above expression has to be one-to-one mapping. This condition can also be defined through the mapping of an infinitesimal volume element

$$J = det(\boldsymbol{F}) \geq 0, \ dv = J\,dV \tag{3.6}$$

where J is the Jacobian determinant, dv and dV are the infinitesimal volume elements in the current and initial configurations respectively. Apart from the mapping of infinitesimal volume and line elements, the mapping of infinitesimal surface area elements between two configurations is given by

$$\boldsymbol{n}da = J\boldsymbol{F}^{-T} \cdot \boldsymbol{N}dA, \tag{3.7}$$

in which $\boldsymbol{N}$ is the outward unit normal vector of the surface element dA in the initial configuration and $\boldsymbol{n}$ is the outward unit normal vector of the surface area element da in the current configuration.

The deformation gradient $\boldsymbol{F}$ describes total deformations, including changes in the shape of body and rigid body rotations. No strains are expected to occur for body with pure rigid body motions. Since deformation gradient never equals zero, it cannot be used as a strain measurement. This limitation can be overcome through polar decomposition of the deformation gradient into pure rotation $\boldsymbol{R}$ and a stretch tensor

$$\boldsymbol{F} = \boldsymbol{RU} = \boldsymbol{VR} \tag{3.8}$$

where U and V are the right and left stretch tensor respectively. Now, we define the Cauchy-Green tensor in the initial configuration

$$C = F^T F = U^T \underbrace{R^T R}_{1} U = U^2 \tag{3.9}$$

Subsequently, an alternative approach to describe the deformation of a body is made through Green-Lagrange strain tensor which is defined in the initial configuration in terms of Cauchy-Green tensor C

$$E = \frac{1}{2}(C - 1) \tag{3.10}$$

Accordingly, in the current configuration, the Euler-Almansi strain tensor is defined as

$$e = \frac{1}{2}(1 - b^{-1}) \tag{3.11}$$

where $b = F F^T = V^2$ is the left Cauchy-Green tensor defined in the current configuration.

Now, the material and spatial velocity gradients are defined to study the deformation rate of the continuum body. The time derivative of the material deformation gradient is defined as the partial derivative of the velocity with respect to the initial configuration

$$\dot{F} = \frac{\partial v\,(X\,,\,t)}{\partial X} \tag{3.12}$$

The time derivative of the spatial deformation gradient is defined as the partial derivative of the velocity with respect to the current configuration

$$l = \frac{\partial v\,(X\,,\,t)}{\partial x} \tag{3.13}$$

Hence, the relation between the spatial and material deformation gradient

$$l = \dot{F} F^{-1} \tag{3.14}$$

Also, the spatial velocity gradient l can be decomposed into a symmetric d and skew-symmetric part w

$$l = d + w \tag{3.15}$$
$$d = \frac{1}{2}(l + l^T) \tag{3.16}$$
$$w = \frac{1}{2}(l - l^T) \tag{3.17}$$

Using Equations (3.14) and (3.16), the time derivative of the Green-Lagrange strain tensor also results from a pull-back of the stretching tensor to the initial configuration.

$$\dot{E} = \frac{d}{dt}\left[\frac{1}{2}(F^T F - 1)\right] = \frac{1}{2}(\dot{F}^T F + F^T \dot{F}) = F^T d F \tag{3.18}$$

Hence, it shows that time derivative is frame indifferent and it results out from the fact that both the deformation gradient and the stretching tensor are objective.

The time derivative of the Euler-Almansi strain tensor is given as

$$\dot{\boldsymbol{e}} = \frac{d}{dt}\left[\frac{1}{2}(1 - \boldsymbol{F}^{-T}\boldsymbol{F}^{-1})\right] = \frac{1}{2}(-\dot{\boldsymbol{F}}^{-T}\boldsymbol{F}^{-1} + \boldsymbol{F}^{-T}\dot{\boldsymbol{F}}^{-1}) = \boldsymbol{d} - \boldsymbol{l}^{T}\boldsymbol{e} - \boldsymbol{e}\boldsymbol{l} \quad (3.19)$$

It does not result from the push-forward of the Green-Lagrange strain tensor and the derivative is not objective. Hence, Lie derivative $\mathfrak{L}$ is introduced to cope with this problem. This derivative consists of three steps, which start with the pull-back to the reference configuration, followed by a material time derivative and then push-forward back to the current configuration. The derivation of the framework is inhibited through this approach and it leads to an objective time derivative. The lie derivative of Euler-Almansi strain tensor is given as

$$\mathfrak{L}(\boldsymbol{e}) = \boldsymbol{F}^{-T}\frac{d}{dt}\left(\boldsymbol{F}^{T}\boldsymbol{e}\boldsymbol{F}\right)\boldsymbol{F}^{-1} = \boldsymbol{F}^{-T}\dot{\boldsymbol{E}}\boldsymbol{F}^{-1} = \boldsymbol{d} \quad (3.20)$$

3.2.2 Stresses

Stress is defined as a physical quantity expressing the force per unit area exerted by a continuum particle on its neighboring particle across an imaginary surface separating them. Different stress tensors are applied in different configurations withing continuum mechanics framework. In current configuration, an important stress definition, known as Cauchy stresses, is given by the Cauchy-theorem (or Cauchy Tetrahedron Theorem) as

$$\boldsymbol{t}\left(\mathbf{X},\,t\right) = \boldsymbol{\sigma}^{T}\left(\mathbf{x},\,t\right)\boldsymbol{n} \quad (3.21)$$

where $\boldsymbol{t}$ is the surface traction vector defined by the forces measured per unit surface area in the current configuration, $\boldsymbol{n}$ is the outward unit normal vectors to the surface. This equation links the force with the Cauchy stress stensor $\boldsymbol{\sigma}$ and its associated outward normal. Nominal or 1$^{\text{st}}$ Piola-Kirchhoff stress tensor which relate both the configurations is defined as

$$\boldsymbol{T}\left(\mathbf{X},\,t\right) = \mathbf{P}^{T}\left(\mathbf{X},\,t\right)\boldsymbol{N}. \quad (3.22)$$

where the traction $\boldsymbol{T}$ denotes the force measured per unit surface area in the reference configuration. Nominal stress or 1$^{\text{st}}$ Piola-Kirchhoff stress tensor $\boldsymbol{P}$ is a two-point unsymmetric tensor and can be related to the Cauchy stress ($\boldsymbol{\sigma}$) through the deformation gradient ($\boldsymbol{F}$) and Jacobian (J) as

$$\boldsymbol{\sigma} = J^{-1}\boldsymbol{P}\boldsymbol{F}^{T} \quad (3.23)$$

The Kirchhoff stress tensor ($\boldsymbol{\tau}$), which is often used in plasticity, is defined in the current configuration and it is related to the Cauchy stress ($\boldsymbol{\sigma}$) through Jacobian (J) as

$$\boldsymbol{\tau} = J\boldsymbol{\sigma}. \quad (3.24)$$

Another measure of stress more widely used is the 2$^{\text{nd}}$ Piola-Kirchhoff stress ($\mathbf{S}$) where both the force and the area are in the reference configuration. This is related to the Cauchy stress ($\boldsymbol{\sigma}$) as

$$\mathbf{S} = \mathbf{F}^{-1}\mathbf{P} = J\mathbf{F}^{-1}\boldsymbol{\sigma}\mathbf{F}^{-T} \quad (3.25)$$

Both Cauchy and 2$^{\text{nd}}$ Piola-Kirchhoff stresses are both symmetric tensors.

3.2.3 Balance Equations

Balance laws define the material independant conservation quantities, which are essential to define the phenomological character of the problem. Balance laws establish the equalities/ inequalities which balance the essential physical quantities (like sources and fluxes) in both solids and fluids. This section discusses the conservation laws pertaining to mass, linear and angular momentum, energy and entropy. Each of these conservation laws can be defined in both Lagrangian and Eulerian frameworks.

Balance of Mass

Consider the body Ω_0 in initial configuration undergoing deformation in a closed system during which the mass has to remain constant. This means that masss is not subjected to any change over time and it is a conserved quantity. Hence, the time derivative of the total mass of a body is zero

$$\dot{m} = \frac{d}{dt} \int_{\Omega_0} \rho_0 dV = \frac{d}{dt} \int_{\Omega} \rho dv = 0, \tag{3.26}$$

where ρ_0 and ρ represent the mass density of the body in the initial and final configurations respectively. Modifying Equation (3.6) and applying to the last term of Equation (3.26),

$$\frac{d\left(\rho J\right)}{dt} = J\left(\dot{\rho} + \rho \operatorname{div} \dot{\boldsymbol{x}}\right) = 0, \tag{3.27}$$

and this leads to the continuity mass equation in its local rate form

$$\dot{\rho} + \rho \operatorname{div} \dot{\boldsymbol{x}} = 0. \tag{3.28}$$

Balance of linear and angulular Momentum

The balance principle of linear momentum states that the time derivative of the linear momentum is equal to the sum of the external forces acting on the body Ω

$$\frac{d}{dt} \int_{\Omega} \rho \dot{\boldsymbol{x}} dv = \int_{\Omega} \rho \boldsymbol{b} dv + \int_{\partial\Omega} \boldsymbol{t} \, da \tag{3.29}$$

where $\boldsymbol{t}$ is the surface traction acting on the boundary $\partial\Omega$ and $\boldsymbol{b}$ is the body force density acting on the volume of the body. Using Equation (3.21) and applying the divergence theorem to the above equation leads to

$$\int_{\Omega} \left(\operatorname{div}\boldsymbol{\sigma} + \boldsymbol{f} - \rho\ddot{\boldsymbol{x}}\right) dv = 0 \tag{3.30}$$

where $\ddot{\boldsymbol{x}}$ is the acceleration field and $\boldsymbol{f}$ is the body force vector per unit volume defined as $\boldsymbol{f} = \rho\boldsymbol{b}$. In local form, the above equation can be written as

$$\operatorname{div}\boldsymbol{\sigma} + \boldsymbol{f} = \rho\ddot{\boldsymbol{x}} \tag{3.31}$$

To fulfill the conservation of linear momentum, Equation (3.31) has to be fulfilled at each point of the continuum.

The angular momentum of a body can be written in the current configuration as

$$\boldsymbol{L} = \int_\Omega \boldsymbol{r} \times \rho\ddot{\boldsymbol{x}}\,dv, \tag{3.32}$$

where $\boldsymbol{r} = (\boldsymbol{x} - \boldsymbol{x}_0)$ gives the distance from a reference fixed vector $\boldsymbol{x}_0$ to any given point $\boldsymbol{x}$. The balance of angular momentum, similar to linear momentum balance, states the change in time of angular momentum which is equal to the sum of the external moments acting on the body Ω

$$\dot{\boldsymbol{L}} = \frac{d}{dt}\int_\Omega \boldsymbol{r} \times \rho\,\dot{\boldsymbol{x}}dv = \int_\Omega \boldsymbol{r} \times \rho\,\boldsymbol{b}dv + \int_{\partial\Omega} \boldsymbol{r} \times \boldsymbol{t}\,da \tag{3.33}$$

The angular momentum is conserved automatically as a result of the symmetry property of the Cauchy stress tensor

$$\boldsymbol{\sigma} = \boldsymbol{\sigma}^T \tag{3.34}$$

Balance of Energy

The First law of thermodynamics or more commonly known as balance of energy states that the change in time of the total amount of energy E_{tot} is equal to the work performed per unit time P^E and the heat supply $\dot{Q}$

$$\dot{E}_{tot} = \int_\Omega \boldsymbol{f} \cdot \dot{\boldsymbol{x}}dv + \int_{\partial\Omega} \boldsymbol{t} \cdot \dot{\boldsymbol{x}}da + \int_\Omega \rho\,rdv - \int_{\partial\Omega} \boldsymbol{q} \cdot \boldsymbol{n}da, \tag{3.35}$$

where $\boldsymbol{q}$ is the heat flux vector and r is the specific heat supply. The above expression can be written in terms of internal energy U and kinetic energy K as

$$E_{tot} = \int_\Omega \frac{1}{2}\rho\dot{\boldsymbol{x}} \cdot \dot{\boldsymbol{x}}dv + \int_\Omega \rho\,\tilde{u}\,dv, \tag{3.36}$$

where $\tilde{u}$ is the specific internal energy. Substituting the above equation in Equation (3.35) leads to

$$\dot{E}_{tot} = \frac{d}{dt}\int_\Omega \rho\left(\frac{1}{2}\dot{\boldsymbol{x}} \cdot \dot{\boldsymbol{x}} + \tilde{u}\right)dv = \int_\Omega (\boldsymbol{f} \cdot \dot{\boldsymbol{x}} + \rho\,r)\,dv + \int_{\partial\Omega} (\boldsymbol{t} \cdot \dot{\boldsymbol{x}} - \boldsymbol{q} \cdot \boldsymbol{n})\,da, \tag{3.37}$$

Using the local form of linear momentum balance (3.31), Cauchy theorem (3.21), symmetry of Cauchy stress tensor $\boldsymbol{\sigma} = \boldsymbol{\sigma}^T$ and applying divergence theorem to Equation (3.37), the local form of first law of thermodynamics can be obtained as

$$\rho\dot{\tilde{u}} = \boldsymbol{\sigma} : \boldsymbol{d} + \rho r - \text{div}\boldsymbol{q} \tag{3.38}$$

Entropy inequality

The balance of entropy or Second law of Thermodynamics or Clausius-Duhem Inequality provides information about the direction of energy transfer. According to this law, for any

isolated system, the total internal entropy always tends to increase or remains constant (for completely reversible process). The entropy production per unit time can be described by the rate of change of entropy and the heat flux divided by the temperature

$$\frac{d}{dt}\int_{\Omega}\eta dv + \int_{\partial\Omega}\frac{\boldsymbol{q}}{\Theta}\cdot\boldsymbol{n}da - \int_{\Omega}\frac{r}{\Theta}dv \geq 0, \tag{3.39}$$

where η is the specific entropy per unit volume in the current configuration and Θ is the temperature.

3.3 Constitutive Model for the coupled problem

Apart from the equations resulting from the balance laws and kinematics, additional relations are needed to describe the behavior of the materials. These relations are known as constitutive equations which provide a functional relationship between kinematic quantities (for instance, strain) and the kinetic quantities such as stress.

3.3.1 Intermediate Configurations

The concept of multiplicative deformation gradient has been introduced for the material modeling at large strains, where the intermediate configurations play an important role. At large strains, the associated strain tensors are not additive anymore, which is in contrast to the small strain regime. The main advantage of this concept is that phenomological observations can be modelled separately considering successive partial deformations. The multiplicative decomposition of deformation gradient into partial deformations B and A, as discussed in the comprehensive review by (LUBARDA, 2004), is given as

$$\boldsymbol{F} = \boldsymbol{F}_B\boldsymbol{F}_A \tag{3.40}$$

Consider a body initially in the reference configuration is subjected to the partial deformation A and the associated deformation gradient $\boldsymbol{F}_A$ and consecutively, it undergoes partial deformation with associated deformation gradient $\boldsymbol{F}_B$. Here, the body, after being subjected to partial deformation A is in intermediate configuration. It is assumed that these partial deformations are taking place separately and not simultaneously. Multiplicative decomposition is not commutative since the order of successive partial deformation matters. But, the Jacobian is commutatively split since it is a scalar quantity.

$$\boldsymbol{F}_B\boldsymbol{F}_A \neq \boldsymbol{F}_A\boldsymbol{F}_B \tag{3.41}$$

$$J = J_B J_A = J_A J_B \tag{3.42}$$

The deformation gradient explains the underlying kinematics for finite strains. Using the concept of multiplicative decomposition of the deformation gradient, the deformation is split into an elastic and an inelastic part,

$$\boldsymbol{F} = \boldsymbol{F}_{el}\boldsymbol{F}_{in} \tag{3.43}$$

Within a generalised Maxwell model, which will be discussed in Section 3.3.3, the above multiplicative decomposition is performed for each Maxwell branch.

3.3.2 Split of the free energy

The stored elastic energy can be represented by the Helmholtz free energy, which is additively decomposable. The multi-physical decomposition of the free energy, which is to be discussed in detail in later sections, is shown here

$$\psi = \psi_M + \psi_{\theta C} \tag{3.44}$$

where, ψ_M is the mechanical part and $\psi_{\theta C}$ is the thermo-chemical part.
The free energy can also be decomposed into isochoric and volumetric part

$$\psi = \psi^{iso}\left(\bar{C}\right) + \psi^{vol}\left(J\right) \tag{3.45}$$

where the isochoric part of the free energy represents the isochoric deformation and it is defined w.r.t the right isochoric Cauchy-Green tensor. This tensor can be obtained by multiplicative split of the deformation gradient into volumetric and isochoric part

$$\boldsymbol{F} = \boldsymbol{F}^{vol}\bar{\boldsymbol{F}} \tag{3.46}$$

Here, the Jacobian of the isochoric part is equal to one. Hence, the volumetric part can be written as $\boldsymbol{F}^{vol} = J^{\frac{1}{3}}\boldsymbol{1}$ and the isochoric part of the right Cauchy-Green tensor can be defined as

$$\bar{\boldsymbol{C}} = J^{-\frac{2}{3}}\boldsymbol{C} \tag{3.47}$$

Considering the generalised Maxwell model (Figure 3.3), the free energy can be formulated in terms of the right Cauchy-Green tensor $\boldsymbol{C}$ as

$$\psi = \psi_\infty\left(\boldsymbol{C}\right) + \sum_{i=1}^{n}\Psi_i^{iso}\left(\boldsymbol{C}_{el_i}\right) \tag{3.48}$$

The elastic part of the free energy function contributes to the total free energy. The right Cauchy-Green tensor $\boldsymbol{C}$ has to be decomposed into elastic $\boldsymbol{C}_{el}$ and inelastic part $\boldsymbol{C}_{in}$. For both the equilibrium and n non-equilibrium parts of the free energy for the generalised Maxwell model, a hyperelastic material has to be applied, for example, Neo-Hookean, Ogden or Mooney-Rivlin model. From Equation (3.9) and Equation (3.43), the elastic right Cauchy-Green tensor can be defined as

$$\boldsymbol{C}_{el} = \boldsymbol{F}_{el}^T\boldsymbol{F}_{el} = \boldsymbol{F}_{in}^{-T}\underbrace{\boldsymbol{F}^T\boldsymbol{F}}_{\boldsymbol{C}}\boldsymbol{F}_{in}^{-1} = \boldsymbol{F}_{in}^{-T}\boldsymbol{C}\boldsymbol{F}_{in}^{-1} \tag{3.49}$$

Applying Clausius-Duhem inequality for the generalized Maxwell model (HARTMANN, 2019) leads to

$$\left(\boldsymbol{S} - 2\rho_0 \frac{\partial \Psi_\infty}{\partial \boldsymbol{C}} - \sum_{i=1}^{n} 2\rho_0 \boldsymbol{F}_{in_i}^{-1} \frac{\partial \Psi_i^{iso}}{\partial \boldsymbol{C}_{el_i}} \boldsymbol{F}_{in_i}^{-T} \right) : \frac{1}{2}\dot{\boldsymbol{C}} - \sum_{i=1}^{n} 2\rho_0 \frac{\partial \Psi_i^{iso}}{\partial \boldsymbol{C}_{el_i}} : \boldsymbol{C}_{el_i} \boldsymbol{L}_{in_i} \geq 0$$

(3.50)

Using the standard Coleman-Noll argument (COLEMAN & NOLL, 1963), the second Piola-Kirchhoff stress can be defined as

$$\boldsymbol{S} = 2\rho_0 \frac{\partial \Psi_\infty}{\partial \boldsymbol{C}} - \sum_{i=1}^{n} 2\rho_0 \boldsymbol{F}_{in_i}^{-1} \frac{\partial \Psi_i^{iso}}{\partial \boldsymbol{C}_{el_i}} \boldsymbol{F}_{in_i}^{-T}$$

(3.51)

The volumetric part of the free energy can be described by the product of the compression modulus K and an associated compressible extension term $w(J)$, resulting in

$$\psi^{vol} = Kw(J)$$

(3.52)

where $w(J)$ can be described according to the approach by Ciarlet (HOLZAPFEL, 2000) as

$$w(J) = \frac{1}{4}\left(J^2 - 1\right) - \frac{1}{2}\ln J$$

(3.53)

The rate of free energy is required to satisfy the second law of thermodynamics as in Section 3.2.3, the rate of free energy is defined w.r.t the right Cauchy-Green tensor as

$$\dot{\Psi} = \frac{\partial \Psi^{iso}}{\partial \bar{\boldsymbol{C}}} : \frac{\partial \bar{\boldsymbol{C}}}{\partial \boldsymbol{C}} : \dot{\boldsymbol{C}} + \frac{\partial \Psi^{vol}}{\partial J}\frac{\partial J}{\partial \boldsymbol{C}} : \dot{\boldsymbol{C}}$$

(3.54)

With

$$\begin{aligned}
\frac{\partial \bar{\boldsymbol{C}}}{\partial \boldsymbol{C}} &= J^{-\frac{2}{3}}\left(\mathbb{I} - \frac{1}{3}\boldsymbol{C} \otimes \boldsymbol{C}^{-1}\right) = \mathbb{P}^T \\
\frac{\partial J}{\partial \boldsymbol{C}} &= \frac{J}{2}\boldsymbol{C}^{-1}
\end{aligned}$$

(3.55)

where $\mathbb{P}$ is the Projection tensor. Considering the formulation for the rate of free energy in terms of the rate of right Cauchy-Green tensor as shown by (IHLEMANN, 2003), the rate of free energy can be reformulated as

$$\dot{\Psi} = \left[\left(\underbrace{\frac{\partial \Psi^{iso}}{\partial \bar{\boldsymbol{C}}}\bar{\boldsymbol{C}}}_{\bar{s}} \right)\boldsymbol{C}^{-1} + \frac{\partial \Psi^{vol}}{\partial J}\frac{J}{2}\boldsymbol{C}^{-1} \right] : \dot{\boldsymbol{C}}$$

(3.56)

Further detailed formulation can be found in (HARTMANN, 2019). In order to incorporate the curing phenomena and the related thermo-chemical process dependencies, an experimentally based thermo-chemical free energy function has been introduced. As discussed in (KOLMEDER ET AL., 2011), the enthalpy is taken as a thermodynamic potential. The enthalpy rate was calculated through DSC experiements. This technique consists of two measurement methods where the absorbed heat powers of the sample and the reference material are measured. And, under nearly stress-free conditions during DSC experiments, the

dependence of enthalpy on stress will be neglected and therefore, only the thermo-chemical part will be considered. From these considerations, the rate of enthalpy is given as

$$\dot{h} = \frac{\partial h}{\partial \Theta} \dot{\Theta} + \frac{\partial h}{\partial \alpha} \dot{\alpha} \tag{3.57}$$

From experiments by (KOLMEDER ET AL., 2011), model of enthalpy linearly dependent on the degree of cure and quadratically dependent on the temperature has been given as

$$h_{\Theta C}(\Theta, \alpha) = h_F(\Theta)(1 - \alpha) + h_S(\Theta)\alpha \tag{3.58}$$

$$h_F(\Theta) = h_{F0} + a_F(\Theta - \Theta_0) + \frac{1}{2}b_F(\Theta - \Theta_0)^2 \tag{3.59}$$

$$h_S(\Theta) = h_{S0} + a_S(\Theta - \Theta_0) + \frac{1}{2}b_S(\Theta - \Theta_0)^2 \tag{3.60}$$

where h_F is the enthalpy of the uncured material and h_S is the enthalpy of the completely cured material. The parameters a_F, a_S, b_F and b_S are determined from DSC experiments. These two temperature dependent enthalpies are defined with respect to the reference temperature. An alternative function proposed by (LANDGRAF, 2015) does not refer to the reference temperature which is defined as

$$h_F(\Theta) = h_{F0} + \frac{1}{2}a_F\Theta^2 + \frac{1}{3}b_F\Theta^3$$
$$h_S(\Theta) = h_{S0} + \frac{1}{2}a_S\Theta^2 + \frac{1}{3}b_S\Theta^3 \tag{3.61}$$

From the definition $\Delta h_{FS} = h_{S0} - h_{F0}$ and using Equation (3.61), the enthalpy (3.58) can be approximated as

$$\begin{aligned} h_{\Theta C}(\Theta, \alpha) = h_{F0} + \Delta h_{FS}\alpha + \left(\frac{1}{2}a_F\Theta^2 + \frac{1}{3}b_F\Theta^3\right)(1 - \alpha) \\ + \left(\frac{1}{2}a_S\Theta^2 + \frac{1}{3}b_S\Theta^3\right)\alpha \end{aligned} \tag{3.62}$$

where all the parameters are determined from DSC experiments.
Also, the thermo-chemical specific heat capacity is obtained from Equation (3.61) as

$$c_{\Theta C} = \frac{\partial h_{\Theta C}}{\partial \Theta} = \left(a_F\Theta + b_F\Theta^2\right)(1 - \alpha) + \left(a_S\Theta + b_S\Theta^2\right)\alpha \tag{3.63}$$

Since no measurements for the material parameters were available, the parameters for the arcylic bone cement by (LANDGRAF, 2015) are used. They are depicted in the Table 3.1.

3.3.3 Constitutive modeling

Photopolymerization process couples the material property evolution and the mechanical deformation. In the beginning, the material is modeled as a viscous melt with the introduction of crosslinks. This liquid state exists before the gel point where the material can carry

Parameter	Δh_{FS}	a_F	b_F	a_S	b_S
Value	-80.95	4.5729×10^{-3}	-1.9600×10^{-6}	4.1210×10^{-3}	-1.6785×10^{-6}
Unit	$[J/g]$	$[J/g.K^2]$	$[J/g.K^3]$	$[J/g.K^2]$	$[J/g.K^3]$

Table 3.1. Fitted material parameters for the thermo-chemical quantities based on DSC experiment for arcylic bone cement,cf. (LANDGRAF, 2015)

short-term load only due to its viscosity. During polymerization process, the polymer chains grow up and crosslink. At this stage, the material passes the gel point and the material is able to carry the long-term load. Due to the progress in the reaction and increase in the cross-linking process, there is increase in crosslink density, thereby resulting in continuous increase in stiffness. These newly formed chains are not affected by the previous deformation and do not contribute to the stress until further deformation (GILLEN, 1988; HOSSAIN ET AL., 2009; WU ET AL., 2018) and hence, these new crosslinks are in a stress-free state. The viscous effects (non-equilibrium behaviors) can be modeled by the multibranch model consisting of series of Maxwell elements in parallel.

The rheological model (see Figure 3.3) consists of a process-dependent Maxwell model with two parts: the equilibrium branches and non-equilibrium branches. The equilibrium branches contribute to the phase evolution and the branches increases with the increase in number of crosslinks (LONG ET AL., 2010). The non-equilibrium branches represent the stress-relaxation model of the polymer. The newly formed crosslinks will be in a stress free state untill the deformation is changed. But, these new crosslinks will hinder the relaxation rate of the old crosslinks and hence, the existing relaxation time will be shifted to a new relaxation time.
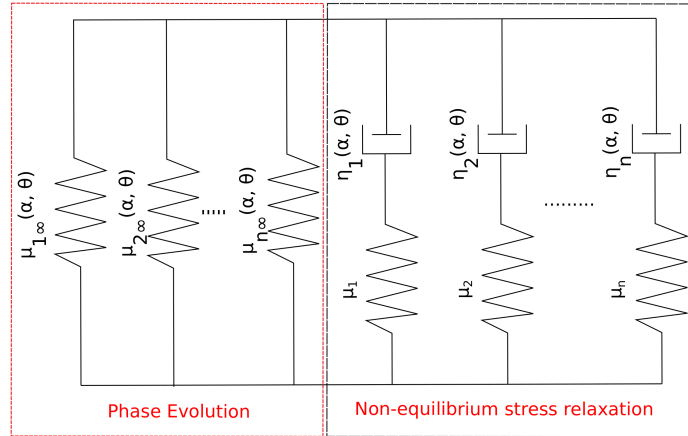

Figure 3.3. Rheological model for viscoelastic mechanical properties modeling

In order to model the large strain curing process, multiplicative decomposition of deformation gradient into mechanical, thermal and chemical part is used, see (LION & HÖFER, 2007)

$$\boldsymbol{F} = \boldsymbol{F}_M \boldsymbol{F}_C \boldsymbol{F}_\Theta \tag{3.64}$$

The mechanical deformation is considered to be completely viscoelastic, $\mathbf{F}_M = \mathbf{F}_M^{ve}$. The deformation within the equilibrium spring can be further split into volumetric and isochoric parts $\mathbf{F}_M^{ve} = \mathbf{F}_M^{ve,vol} \bar{\mathbf{F}}_M^{ve}$. Both these parts share the elastic stored energy. For each Maxwell branch, the viscoelastic deformation is split into elastic ($\mathbf{F}_{M_{el_i}}^{ve}$) and inelastic ($\mathbf{F}_{M_{in_i}}^{ve}$) parts. The inelastic part is the dissipative viscous deformation in the dashpot.

Isotropic volumetric approaches for thermal expansion and chemical shrinkage are used

$$\boldsymbol{F}_C = g\,(\alpha)^{1/3}\,\mathbf{1}, \ \boldsymbol{F}_\Theta = \phi\,(\alpha,\,\Theta)^{1/3}\,\mathbf{1} \tag{3.65}$$

where $g\,(\alpha)$ and $\phi\,(\alpha,\,\Theta)$ are the scalar functions which represent thermal expansion and chemical shrinkage in principal directions. These volumetric changes are determined as

$$
\begin{aligned}
g\,(\alpha) &= 1 + \beta_c \alpha \text{ with } \beta_c \leq 0 \\
\phi\,(\alpha,\,\Theta) &= 1 + \left[(1-\alpha)\beta_{\Theta_f} + \alpha\beta_{\Theta_s}\right](\Theta - \Theta_0) \text{ with } \beta_{\Theta_f},\,\beta_{\Theta_s} \geq 0
\end{aligned} \tag{3.66}
$$

Here, we assume that chemical shrinkage and thermal expansion are not directly coupled. Hence, thermal expansion depends on temperature difference with respect to reference temperature and the degree of cure α. It can be linked to the Temperature model through the temperature difference and Photopolymerization model through degree of cure as shown in Figure 3.1. And, the chemical shrinkage solely depends on the degree of cure α and the chemical shrinkage parameter, β_c. The values of chemical shrinkage and thermal expansion parameters are depicted in Table 6.1.

The material generally shrinks upon solidification. Solidification is the build-up of mechanical properties in the material, which represents the evolution of cross-linking of polymer chains and introducing solid-like behavior during photopolymerization process. However, transient phenomena involving transformation from a liquid into a solid will be ignored here. In order to model the solidification and the temperature-dependent stiffness of the material, the shear modulus of the equilibrium spring is dependent on both degree of cure and temperature. For the purpose of simplification in our model, in order the exhibit the linear growth, the shear modulus is dependent on the degree of cure only, cf. (HOSSAIN ET AL., 2009),

$$\mu_\alpha(\alpha) = p_{\alpha_1}^\mu(1 - \alpha) + p_{\alpha_2}^\mu \alpha \tag{3.67}$$

where the parameters $p_{\alpha_1}^\mu$ and $p_{\alpha_2}^\mu$ are the temperature independent shear moduli for the uncured and fully cured material respectively. To model the fluid-like behavior during the beginning of photopolymerization process, the cure-dependent yield stress, σ_y is defined.

Within each Maxwell element, the viscoelastic split of deformation gradient can be related as

$$\dot{\boldsymbol{C}}_{M_{in_i}}^{ve} = \frac{1}{\tau(\alpha,\Theta)}\left(\boldsymbol{C}_M^{ve} - \boldsymbol{C}_{M_{in_i}}^{ve}\right) \tag{3.68}$$

where τ is the relaxation time, which is cure dependent.

Split of stress

From the definition of the Second Piola Kirchhoff stress tensor (3.25) and using the multiplicative split of the deformation gradient (3.64) leads to the representation

$$\boldsymbol{S} = J_M J_C J_\Theta \boldsymbol{F}_\Theta^{-1} \boldsymbol{F}_C^{-1} \boldsymbol{F}_M^{-1} \boldsymbol{\sigma} \boldsymbol{F}_M^{-T} \boldsymbol{F}_C^{-T} \boldsymbol{F}_\Theta^{-T} \tag{3.69}$$

where the mechanical second Piola-Kirchhoff stress tensor is defined as $\boldsymbol{S}_M = J_M \boldsymbol{F}_M^{-1} \boldsymbol{\sigma} \boldsymbol{F}_M^{-T}$, which is the pull back of the mechanical second Piola-Kirchhoff stress tensor defined in the intermediate thermo-chemical configuration and Equation (3.69) is represented as

$$\boldsymbol{S} = J_C J_\Theta \boldsymbol{F}_\Theta^{-1} \boldsymbol{F}_C^{-1} \boldsymbol{S}_M \boldsymbol{F}_C^{-T} \boldsymbol{F}_\Theta^{-T} \tag{3.70}$$

Now, the Green-Lagrange strain tensor (3.10) can be represented as

$$\boldsymbol{E} = \frac{1}{2} \left(\boldsymbol{F}_\Theta^T \boldsymbol{F}_C^T \boldsymbol{F}_M^T \boldsymbol{F}_M \boldsymbol{F}_C \boldsymbol{F}_\Theta - \boldsymbol{1} \right) \tag{3.71}$$

and the corresponding mechanical right Cauchy-Green tensor and Green-Lagrange strain tensor is given as

$$\boldsymbol{C}_M = \boldsymbol{F}_M^T \boldsymbol{F}_M \tag{3.72}$$

$$\boldsymbol{E}_M = \frac{1}{2} \left(\boldsymbol{C}_M - \boldsymbol{1} \right) \tag{3.73}$$

We can define the rate of Green-Lagrange strain tensor as

$$\begin{aligned}
\dot{\boldsymbol{E}} = \frac{1}{2} \Big(&\dot{\boldsymbol{F}}_\Theta^T \boldsymbol{F}_C^T \boldsymbol{C}_M \boldsymbol{F}_C \boldsymbol{F}_\Theta + \boldsymbol{F}_\Theta^T \dot{\boldsymbol{F}}_C^T \boldsymbol{C}_M \boldsymbol{F}_C \boldsymbol{F}_\Theta + \boldsymbol{F}_\Theta^T \boldsymbol{F}_C^T \dot{\boldsymbol{C}}_M \boldsymbol{F}_C \boldsymbol{F}_\Theta \\
&+ \boldsymbol{F}_\Theta^T \boldsymbol{F}_C^T \boldsymbol{C}_M \dot{\boldsymbol{F}}_C \boldsymbol{F}_\Theta + \boldsymbol{F}_\Theta^T \boldsymbol{F}_C^T \boldsymbol{C}_M \boldsymbol{F}_C \dot{\boldsymbol{F}}_\Theta \Big)
\end{aligned} \tag{3.74}$$

Further, considering the implementation of (HARTMANN, 2019), thermal and chemical velocity gradients are represented as

$$\begin{aligned}
\boldsymbol{L}_\Theta &= \dot{\boldsymbol{F}}_\Theta \boldsymbol{F}_\Theta^{-1} = \left(\frac{\partial \varphi}{\partial \Theta} \dot{\Theta} + \frac{\partial \varphi}{\partial \alpha} \dot{\alpha} \right) \boldsymbol{1} \frac{1}{3\varphi} \boldsymbol{1} = \frac{1}{3\varphi} \frac{\partial \varphi}{\partial \Theta} \dot{\Theta} \boldsymbol{1} \\
\boldsymbol{L}_C &= \dot{\boldsymbol{F}}_C \boldsymbol{F}_C^{-1} = \left(\frac{\partial g}{\partial \Theta} \dot{\Theta} + \frac{\partial g}{\partial \alpha} \dot{\alpha} \right) \boldsymbol{1} \frac{1}{3g} \boldsymbol{1} = \frac{1}{3g} \frac{\partial g}{\partial \alpha} \dot{\alpha} \boldsymbol{1}
\end{aligned} \tag{3.75}$$

Now, the stress power $\boldsymbol{P}_{int} = \boldsymbol{S} : \dot{\boldsymbol{E}}$ can be split additively into mechanical, thermal and chemical parts as

$$\boldsymbol{S} : \dot{\boldsymbol{E}} = g\varphi \left(\underbrace{\boldsymbol{S}_M : \dot{\boldsymbol{E}}_M}_{mechanical} + \underbrace{\boldsymbol{C}_M \boldsymbol{S}_M : \left(\boldsymbol{F}_C \boldsymbol{L}_\Theta \boldsymbol{F}_C^{-1} \right)}_{thermal} + \underbrace{\boldsymbol{C}_M \boldsymbol{S}_M : \boldsymbol{L}_C}_{chemical} \right) \tag{3.76}$$

From Equation (3.76) and inserting $\boldsymbol{L}_\Theta$ from Equation (3.75), the thermal stress power can be represented as

$$\boldsymbol{C}_M \boldsymbol{S}_M : \left(\boldsymbol{F}_C \boldsymbol{L}_\Theta \boldsymbol{F}_C^{-1}\right) = \text{tr}\left(\left(\boldsymbol{F}_M^T \boldsymbol{F}_M\right)\left(J_M \boldsymbol{F}_M^{-1} \boldsymbol{\sigma} \boldsymbol{F}_M^{-T}\right) \boldsymbol{F}_C \left(\frac{1}{3\varphi}\frac{\partial\varphi}{\partial\Theta}\dot{\Theta}\mathbf{1}\right)\boldsymbol{F}_C^{-1}\right)$$

$$= \frac{J_M \text{tr}(\boldsymbol{\sigma})}{3\varphi}\frac{\partial\varphi}{\partial\Theta}\dot{\Theta} \tag{3.77}$$

and the chemical stress power as

$$\boldsymbol{C}_M \boldsymbol{S}_M : \boldsymbol{L}_C = \text{tr}\left(\left(\boldsymbol{F}_M^T \boldsymbol{F}_M\right)\left(J_M \boldsymbol{F}_M^{-1}\boldsymbol{\sigma}\boldsymbol{F}_M^{-T}\right)\frac{1}{3g}\frac{\partial g}{\partial\alpha}\dot{\alpha}\right) = \frac{J_M \text{tr}(\boldsymbol{\sigma})}{3g}\frac{\partial g}{\partial\alpha}\dot{\alpha} \tag{3.78}$$

The Causius Duhem Inequality as given by (HARTMANN, 2019) can be rewritten as

$$-\rho_0\dot{\Psi} + g\varphi\left(\frac{1}{2}\mathbf{S}_M : \dot{\mathbf{C}}_M + \frac{J_M \text{tr}(\boldsymbol{\sigma})}{3\varphi}\frac{\partial g}{\partial\alpha}\dot{\alpha} + \frac{J_M \text{tr}(\boldsymbol{\sigma})}{3\varphi}\frac{\partial\varphi}{\partial\Theta}\dot{\Theta}\right)$$

$$- \rho_0 s\dot{\Theta} - \frac{1}{\Theta}\cdot\text{Grad}\,\Theta \geq 0 \tag{3.79}$$

Free Energy

To incorporate the curing phenomena and the associated thermo-chemical process dependencies, the Helmholtz free energy function is decomposed additively into mechanical and thermo-chemical part

$$\Psi = \Psi_M + \Psi_{\Theta C} \tag{3.80}$$

The free energy for the mechanical part can be decomposed for one equilibrium and n non-equilibrium parts of the Maxwell model as

$$\Psi = \Psi_{M_\infty} + \sum_{i=1}^{n}\Psi_{M_i} \tag{3.81}$$

The equilibrium part of the mechanical free energy function can be further divided into process-dependent isochoric part and process independent volumetric part

$$\rho_0\Psi_{M_\infty} = \rho_0\Psi_{M_\infty}^{iso}\left(\bar{\boldsymbol{C}}_M, \Theta\right) + \rho_0\Psi_{M_\infty}^{vol}\left(J_M\right)$$

$$= \left[-f_\Theta\int_{-\infty}^{t}\mu_\alpha(s)\left(\frac{d}{ds}\bar{\boldsymbol{C}}_M^{-1}(s)\right)ds\right] : \bar{\boldsymbol{C}}_M + K\omega(J_M) \tag{3.82}$$

where f_Θ is the temperature dependent part of the shear modulus.

The formulation of first term, $\rho_0\Psi_{M_\infty}^{iso}\left(\bar{\boldsymbol{C}}_M, \Theta\right)$ in Equation (3.82) has been discussed in detail in (LANDGRAF ET AL., 2014).

The free energy of the individual Maxwell element can be represented by a Neo-Hookean model

$$\rho_0 \Psi_{M_i}(\boldsymbol{C}_{M_{el_i}}) = \frac{\mu_i}{2} \left(I_{C_{M_{el_i}}} - 3 \right) \tag{3.83}$$

Similarly, the experimentally determined thermo-chemical free energy function, as discussed in Section 3.3.2, is given as

$$\Psi_{\Theta C}(\alpha, \Theta) = h_{F0} + \Delta h_{FS}\alpha - \left(\frac{1}{2} a_F \Theta^2 + \frac{1}{6} b_F \Theta^3 \right)(1 - \alpha) \\ - \left(\frac{1}{2} a_S \Theta^2 + \frac{1}{6} b_S \Theta^3 \right) \alpha \tag{3.84}$$

where the thermo-chemical quantities based on DSC experiments for the arcylic bone cement have been used , which were obtained from (HARTMANN, 2019). They are depicted in Table 3.1. Hence, the total free energy is given by

$$\rho_0 \Psi = \rho_0 \Psi_{M_\infty}^{iso}(\bar{\boldsymbol{C}}_M, \Theta) + \rho_0 \Psi_{M_\infty}^{vol}(J_M) + \sum_{i=1}^{n} \rho_0 \Psi_{M_i}(\boldsymbol{C}_{M_{el_i}}) + \rho_0 \Psi_{\Theta C}(\alpha, \Theta) \tag{3.85}$$

By fulfilling the condition of Clausius-Duhem inequality from Equation (3.79) and considering the dependencies Equation (3.56) and Equation (3.50), the mechanical second Piola-Kirchhoff stress tensor is given by

$$\boldsymbol{S}_M = \frac{2}{g\varphi} \left[\left(\frac{\partial \rho_0 \Psi_{M_\infty}^{iso}}{\partial \bar{\boldsymbol{C}}_M} \bar{\boldsymbol{C}}_M \right)_{dev} \boldsymbol{C}_M^{-1} + \frac{\partial \rho_0 \Psi_{M_\infty}^{vol}}{\partial J_M} \frac{J_M}{2} \boldsymbol{C}_M^{-1} \right. \\ \left. + \sum_{i=1}^{n} \boldsymbol{F}_{M_{in_i}}^{-1} \frac{\partial \rho_0 \Psi_{M_i}}{\partial \boldsymbol{C}_{M_{el_i}}} : \mathbb{P}^T \boldsymbol{F}_{M_{in_i}}^{-T} \right] \tag{3.86}$$

and the entropy is obtained as

$$s = \frac{g J_M \operatorname{tr}(\boldsymbol{\sigma})}{3\rho_0} \frac{\partial \varphi}{\partial \Theta} - \frac{\partial \Psi_{M_\infty}^{iso}}{\partial \Theta} - \frac{\partial \Psi_{\Theta C}}{\partial \Theta} \tag{3.87}$$

Here, $\mathbf{S}_M$ in Equation (3.86) can be represented as $\mathbf{S}_\infty^{vol} + \mathbf{S}_\infty^{iso} + \mathbf{S}^{ve}$, where $\mathbf{S}_\infty^{vol}$ and $\mathbf{S}_\infty^{iso}$ are the isochoric and viscous components of the equilibrium part and $\mathbf{S}^{ve}$ represent the viscous part.

The isochoric and volumetric part of the mechanical second Piola-Kirchhoff stress tensor can be defined from Equation (3.82), where $\omega(J)$ can be defined as given by (CIARLET, 1988)

$$\omega(J) = \frac{1}{4} \left(J^2 - 1 \right) - \frac{1}{2} \ln J \tag{3.88}$$

The associated derivatives of the individual parts of the Equation (3.86) can be defined as

$$\left(\frac{\partial 2\rho_0 \Psi_{M_\infty}^{iso}}{\partial \bar{\boldsymbol{C}}_M}\bar{\boldsymbol{C}}_M\right)_{dev}\boldsymbol{C}_M^{-1} = J_M^{-\frac{2}{3}}\mu_\infty\left(\mathbf{I} - \frac{1}{3}\boldsymbol{C}_M^{-1}\operatorname{tr}(\boldsymbol{C}_M)\right)$$

$$\frac{\partial 2\rho_0\Psi_{M_\infty}^{vol}}{\partial J_M}\frac{J_M}{2}\boldsymbol{C}_M^{-1} = \frac{K}{2}\left(J_M^2 - 1\right)\boldsymbol{C}_M^{-1}$$

$$\sum_{i=1}^{n}\boldsymbol{F}_{M_{in_i}}^{-1}\frac{\partial \rho_0\Psi_{M_i}}{\partial \boldsymbol{C}_{M_{el_i}}} : \mathbb{P}^T\boldsymbol{F}_{M_{in_i}}^{-T} = J_M^{-\frac{2}{3}}\mu_i\left(\boldsymbol{C}_{M_{in_i}}^{-1} - \frac{1}{3}\boldsymbol{C}_M^{-1}\operatorname{tr}(\boldsymbol{C}_M\boldsymbol{C}_{M_{in_i}}^{-1})\right) \quad (3.89)$$

Energy Equation

Using the split of the stress power as in Equation (3.76) and considering the first law of thermodynamics, the energy equation can be derived as

$$\rho_0\dot{\Theta}s + \rho_0\Theta\dot{s} + \rho_0\dot{\Psi} = g\varphi\left(\frac{1}{2}\boldsymbol{S}_M : \dot{\boldsymbol{C}}_M + \frac{J_M\operatorname{tr}(\boldsymbol{\sigma})}{3\varphi}\frac{\partial g}{\partial \alpha}\dot{\alpha} + \frac{J_M\operatorname{tr}(\boldsymbol{\sigma})}{3\varphi}\frac{\partial\varphi}{\partial\Theta}\dot{\Theta}\right)$$
$$- \operatorname{Div}Q + \mathrm{E}_{Laser}$$
$$(3.90)$$

From the Fourier's law of heat conduction, the divergence of the heat flow vector is represented as

$$\operatorname{Div}Q = \operatorname{Div}\left(-\kappa Grad\Theta\right) = -\kappa\operatorname{Div}\left(\operatorname{Grad}\Theta\right) = -\kappa\Delta\Theta \quad (3.91)$$

where κ is the thermal conductivity coefficient.
The specific heat capacity is defined as

$$c_p\left(\alpha, \Theta\right) = \underbrace{-\Theta\frac{\partial^2\Psi_{\Theta C}}{\partial\Theta^2}}_{c_{\Theta C}} - \underbrace{\frac{\partial^2\Psi_{M_\infty}^{iso}}{\partial\Theta^2}}_{c_M} \approx c_{\Theta C} \quad (3.92)$$

which is approximated by its thermo-chemical part and a detailed formulation can be found in (HARTMANN, 2019).
Using the energy equation (3.90), entropy rate from the entropy definition (3.87) and the definition for specific heat capacity (3.92), the energy equation can be defined as

$$c_{\Theta C}\dot{\Theta} = \left(\frac{\varphi J_M\operatorname{tr}(\boldsymbol{\sigma})}{3\rho_0}\frac{\partial g}{\partial\alpha} - \frac{\partial\Psi_{\Theta C}}{\partial\Theta} - \Theta\frac{J_M\operatorname{tr}(\boldsymbol{\sigma})}{3\rho_0}\frac{\partial\varphi}{\partial\Theta}\frac{\partial g}{\partial\alpha} + \frac{\partial^2\Psi_{\Theta C}}{\partial\Theta^2}\right)\dot{\alpha}$$
$$- \Theta\left(\frac{gJ_M\operatorname{tr}(\dot{\boldsymbol{\sigma}})}{3\rho_0}\frac{\partial\varphi}{\partial\Theta}\right) + \frac{1}{\rho_0}\kappa\Delta\Theta + \mathrm{E}_{Laser}$$
$$(3.93)$$

where $c_{\Theta C}$ is defined in Equation (3.63) and rate of the trace of Cauchy stress tensor $\operatorname{tr}(\dot{\boldsymbol{\sigma}})$ is obtained from the entropy rate of the entropy equation (3.87). The stress rates are not directly computed within each time step since small stress rates are expected during the printing process. The rate of the trace of Cauchy stress tensor is computed through the explicit computation as

$$\mathrm{tr}(\dot{\boldsymbol{\sigma}}^{n+1}) = \frac{\mathrm{tr}(\boldsymbol{\sigma}^{n+1}) - \mathrm{tr}(\boldsymbol{\sigma}^n)}{\Delta t} \tag{3.94}$$

Here is a summary of the equations we obtained for the simulation of Stereolithography process

Cure reaction rate using Arrhenius approach

$$\dot{\alpha}(\Theta) = [A_1(\Theta) + A_2(\Theta)\alpha^m] (1-\alpha)^n$$

$$A_1(\Theta) = A_{c1}e^{-\frac{B_1}{\Theta}} \text{ and } A_2(\Theta) = A_{c2}e^{-\frac{B_2}{\Theta}}$$

Material parameters for cure reaction rate are given in Table 6.2.

Functions for thermal expansion and chemical shrinkage

$$g(\alpha) = 1 + \beta_c\alpha \text{ with } \beta_c \leq 0$$

$$\phi(\alpha, \Theta) = 1 + \left[(1-\alpha)\beta_{\Theta_f} + \alpha\beta_{\Theta_s}\right](\Theta - \Theta_0) \text{ with } \beta_{\Theta_f}, \beta_{\Theta_s} \geq 0$$

The material parameters are given in Table 6.1.

Total deformation gradient

$$\boldsymbol{F} = \boldsymbol{F}_M\boldsymbol{F}_C\boldsymbol{F}_\Theta$$

$$\boldsymbol{F}_C = g(\alpha)^{1/3}\,\mathbf{1}, \ \boldsymbol{F}_\Theta = \phi(\alpha, \Theta)^{1/3}\,\mathbf{1}$$

Mechanical right Cauchy-Green tensor

$$\boldsymbol{F}_M = \boldsymbol{F}\boldsymbol{F}_\Theta^{-1}\boldsymbol{F}_C^{-1}$$

$$\boldsymbol{C}_M = \boldsymbol{F}_M^T\boldsymbol{F}_M$$

Shear modulus of the equilibrium spring

$$\mu_\alpha(\alpha) = p_{\alpha_1}^\mu(1-\alpha) + p_{\alpha_2}^\mu\alpha$$

Evolution equation for elastic-inelastic split of viscoelastic deformation gradient within the Maxwell branch

$$\dot{\boldsymbol{C}}_{M_{in_i}}^{ve} = \frac{1}{\tau(\alpha)}\left(\boldsymbol{C}_M^{ve} - \boldsymbol{C}_{M_{in_i}}^{ve}\right)$$

Mechanical second Piola-Kirchoff stress tensor

$$\boldsymbol{S}_M = \frac{2}{g\varphi}\left[\left(\frac{\partial\rho_0\Psi_{M_\infty}^{iso}}{\partial\bar{\boldsymbol{C}}_M}\bar{\boldsymbol{C}}_M\right)_{dev}\boldsymbol{C}_M^{-1} + \frac{\partial\rho_0\Psi_{M_\infty}^{vol}}{\partial J_M}\frac{J_M}{2}\boldsymbol{C}_M^{-1}\right.$$

$$\left. + \sum_{i=1}^n \boldsymbol{F}_{M_{in_i}}^{-1}\frac{\partial\rho_0\Psi_{M_i}}{\partial\boldsymbol{C}_{M_{el_i}}} : \mathbb{P}^T\boldsymbol{F}_{M_{in_i}}^{-T}\right],$$

$$\left(\frac{\partial 2\rho_0\Psi_{M_\infty}^{iso}}{\partial\bar{\boldsymbol{C}}_M}\bar{\boldsymbol{C}}_M\right)_{dev}\boldsymbol{C}_M^{-1} = J_M^{-\frac{2}{3}}\mu_\infty\left(\mathbf{I} - \frac{1}{3}\boldsymbol{C}_M^{-1}\mathrm{tr}(\boldsymbol{C}_M)\right)$$

$$\frac{\partial 2\rho_0\Psi_{M_\infty}^{vol}}{\partial J_M}\frac{J_M}{2}\boldsymbol{C}_M^{-1} = \frac{K}{2}\left(J_M^2 - 1\right)\boldsymbol{C}_M^{-1}$$

$$\sum_{i=1}^{n} \boldsymbol{F}_{M_{in_i}}^{-1} \frac{\partial \rho_0 \Psi_{M_i}}{\partial \boldsymbol{C}_{M_{el_i}}} : \mathbb{P}^T \boldsymbol{F}_{M_{in_i}}^{-T} = J_M^{-\frac{2}{3}} \mu_i \left(\boldsymbol{C}_{M_{in_i}}^{-1} - \frac{1}{3} \boldsymbol{C}_M^{-1} \operatorname{tr}(\boldsymbol{C}_M \boldsymbol{C}_{M_{in_i}}^{-1}) \right)$$

Thermo-chemical free energy function

$$\Psi_{\Theta C}(\alpha, \Theta) = h_{F0} + \Delta h_{FS}\alpha - \left(\frac{1}{2} a_F \Theta^2 + \frac{1}{6} b_F \Theta^3 \right)(1 - \alpha)$$

$$- \left(\frac{1}{2} a_S \Theta^2 + \frac{1}{6} b_S \Theta^3 \right)\alpha$$

Thermo-chemical specific heat capacity

$$c_{\Theta C} = \left(a_F \Theta + b_F \Theta^2 \right)(1 - \alpha) + \left(a_S \Theta + b_S \Theta^2 \right)\alpha$$

Energy Equation

$$c_{\Theta C}\dot{\Theta} = \left(\frac{\varphi J_M \operatorname{tr}(\boldsymbol{\sigma})}{3\rho_0} \frac{\partial g}{\partial \alpha} - \frac{\partial \Psi_{\Theta C}}{\partial \Theta} - \Theta \frac{J_M \operatorname{tr}(\boldsymbol{\sigma})}{3\rho_0} \frac{\partial \varphi}{\partial \Theta} \frac{\partial g}{\partial \alpha} + \frac{\partial^2 \Psi_{\Theta C}}{\partial \Theta^2} \right)\dot{\alpha}$$

$$- \Theta \left(\frac{g J_M \operatorname{tr}(\dot{\boldsymbol{\sigma}})}{3\rho_0} \frac{\partial \varphi}{\partial \Theta} \right) + \frac{1}{\rho_0} \kappa \Delta \Theta + \mathrm{E}_{Laser}$$

This chapter proposes a coupled solution framework which captures the distinct physical phenomena during the photopolymerization process. The proposed framework integrates the coupled effect of four physical mechanisms: 1. light propagation through the resin; 2. conversion (degree of cure evolution) of the resin; 3. evolution of mechanical properties during solidification and 4. thermal effects. The developed numerical framework provides novel insights through a direct coupling between the polymerization kinetics and the temperature evolution, to the build-up residual stresses and mechanical properties.

Chapter 4

Optimal Transportation Meshfree Algorithm

Processes involving large deformations, such as Additive Manufacturing or cutting, present a challenge while modeling with standard approximation tools like the Finite Element Method. If the Lagrangian description is used, these large deformations can result in severe mesh distortions. In this case adaptive remeshing procedures and mapping of state variables from one configuration to another are required. Inefficient computations and accumulated numerical errors can result. Alternatively, meshfree methods seem quite adapted to such simulations. For instance, the Smoothed Particle Hydrodynamics (SPH) has shown big potential. A more recent solution scheme is the Optimal Transportation Meshfree (OTM) method. This method is integrated with local maximum entropy (LME) meshfree interpolation (ARROYO & ORTIZ, 2006) and material point sampling method (SULSKY ET AL., 1994; WESSELS ET AL., 2019, 2018). The advantage of the OTM method is the similar transition of the FEM to a meshfree method.

The Optimal Transportation Meshfree (OTM) method is a discretization scheme in the framework of an Updated Lagrangian formulation which can be used for both solid and fluid flow simulations based on (LI ET AL., 2010). The OTM method can be viewed as an evolution of the finite element method because the spatial domain under investigation is discretized by two types of points. The material points are used as integration points, where quantities like stress, strain, density, etc., are determined. At the nodal points, the primary variables are computed by solving discretized equations of motion. The connectivity between nodes and material points during the computation is established by a search algorithm: the nodes associated with a material point form its support domain, whose shape is in general arbitrary. The material point values are determined with the help of basis functions. In general, maximum entropy shape functions (ARROYO & ORTIZ, 2006) are used.

The OTM method is motivated by the Optimal Transportation Theory. The kinetic energy term is treated by the Optimal Transportation Theory within the formulation of Hamilton's principle. For each nodal point, the Euler-Lagrangian equations are derived after time and spatial discretizations. Since OTM method have some shortcomings, the stabilized formulation due to (WEISSENFELS & WRIGGERS, 2018) is used. Within an Updated Lagrangian framework, the stabilized formulation is derived from the weak form. The whole algorithm

is sketched in Algorithm 1.

4.1 The Weak Form

First of all, initial boundary value problem (IBVP) is specified to solve the Partial Differential Equations (PDEs). Using the balance law of linear momentum formulated with respect to the current configuration, the deformation of the body is

$$\rho \ddot{\boldsymbol{u}} - \rho \hat{\boldsymbol{b}} - \operatorname{div} \boldsymbol{\sigma} = 0, \tag{4.1}$$

where the displacements $\boldsymbol{u}$ are the primary variables. The Cauchy stress tensor, specific body force and density correspond to $\boldsymbol{\sigma}$, $\hat{\boldsymbol{b}}$ and ρ respectively. The Dirichlet and Neumann boundary conditions and the initial conditions for the body are given respectively by

$$\boldsymbol{u} - \hat{\boldsymbol{u}} = 0 \qquad \text{on } \partial_u \Omega, \tag{4.2}$$

$$\boldsymbol{t} - \hat{\boldsymbol{t}} = 0 \qquad \text{on } \partial_t \Omega, \tag{4.3}$$

$$\boldsymbol{u}(t = 0) = \boldsymbol{u}_0 \qquad \text{in } \Omega, \tag{4.4}$$

$$\boldsymbol{v}(t = 0) = \boldsymbol{v}_0 \qquad \text{in } \Omega. \tag{4.5}$$

As shown in (WEISSENFELS & WRIGGERS, 2018), the OTM method can also be derived from the weak form. In case the formulation is made with respect to the current configuration, the virtual work at the boundary balances with virtual work inside of the body and the inertia term.

$$\int_\Omega \delta \boldsymbol{u} \cdot \rho \ddot{\boldsymbol{u}} dv + \int_\Omega \operatorname{grad} \delta \boldsymbol{u} : \boldsymbol{\sigma} dv = \int_\Omega \delta \boldsymbol{u} \rho \cdot \hat{\mathbf{b}} dv + \int_{\partial_t \Omega} \delta \boldsymbol{u} \cdot \hat{\mathbf{t}} da, \tag{4.6}$$

4.2 Spatial discretization

The spatial domain under investigation (as shown in Figure 4.1) is discretized by two sets of points, with different functions: Nodes and material points. The support domain is defined as the domain around each material point, containing nearest nodes in its neighborhood (Figure 4.1). This domain is updated at every computation step by applying a suitable search algorithm and hence, the shape functions $N_I(\boldsymbol{x}_{pn})$ are also continuously updated. At each material point, the test function and the displacements are approximated through shape functions $N_I(\boldsymbol{x}_{pn})$ and nodal values within its support domain

$$\boldsymbol{u}_p(\mathbf{x}_{pn}) = \sum_{I=1}^{n_{np}} N_I(\boldsymbol{x}_{pn})\boldsymbol{u}_I, \ \delta \boldsymbol{u}_p = \sum_{I=1}^{n_{np}} N_I(\boldsymbol{x}_{pn})\delta \boldsymbol{u}_I, \ \operatorname{grad}(\delta \boldsymbol{u}_p) = \sum_{I=1}^{n_{np}} \boldsymbol{B}_I(\boldsymbol{x}_{pn})\delta \boldsymbol{u}_I,$$
$$\tag{4.7}$$

where n_{np} specifies the number of nodes in the support domain of each material point at current computation step. The matrix $\boldsymbol{B}_I(\boldsymbol{x}_{pn})$ contains the derivatives of shape functions at node I. In contrast to the FEM, overlapping of support domains is allowed in OTM method.

In problems of large deformations, non-admissible nodal distributions can be eliminated by the update of support domains at every time or load step.

Using (4.7), the approximation of (4.6) can be transformed into an algebraic equation using an assembly procedure symbolized by operator $A_{p=1}^{n_{mp}}$

$$\left[A_{p=1}^{n_{mp}} \sum_{I}^{n_{np}} \sum_{J}^{n_{np}} N_I(\boldsymbol{x}_p)\mathbf{1}N_J(\boldsymbol{x}_p)m_p \right] \ddot{\boldsymbol{u}} = A_{p=1}^{n_{mp}} \sum_{I}^{n_{np}} \left[N_I(\boldsymbol{x}_p)\hat{\boldsymbol{b}}_p m_p - \boldsymbol{B}_I(\boldsymbol{x}_p)\boldsymbol{\sigma}_p v_p \right]. \tag{4.8}$$

where $\ddot{\boldsymbol{u}}$ is the global nodal acceleration vector, n_{mp} is the total number of material points in the body, m_p is the mass at the material point p and v_p is its volume in the current configuration. In order to guarantee that the conservation of the mass during the computation, the mass of a material point is assumed to be constant. Equation (4.8) can be abbreviated as

$$\boldsymbol{M}\ddot{\boldsymbol{u}} = \boldsymbol{f} - \boldsymbol{P}(\boldsymbol{u}) \tag{4.9}$$

where $\boldsymbol{M}$ denotes the consistent mass matrix, $\boldsymbol{f}$ contains the applied body forces and $\boldsymbol{P}$ stands for the internal force vector. The solution of this discretized dynamic equation makes use of the updated nodal point data and the material point data.

The coupled solution framework (as described in Chapter 3) to capture the physical phenomena during photopolymerization process have been incorporated within the OTM framework. Starting with the cure reaction rate as in Equation (2.5), its meshfree discretization involves evaluation of degree of cure at the material point

$$\dot{\alpha}_{pn} = \left[A_{1,pn} + A_{2,pn}\alpha_{pn-1}^m \right] (1 - \alpha_{pn-1})^n \tag{4.10}$$

where α_{pn-1} is the degree of cure at previous time step. The temperature which is defined at a node can be written in the discretized form at a material point as

$$\Theta_{pn} = \sum_{I}^{n_{np}} N_I(\boldsymbol{x}_{pn})\Theta_{In} \tag{4.11}$$

which is utilised to compute the temperature dependent rate constants $A_{1,pn}$ and $A_{2,pn}$ of the cure reaction rate in Equation (4.10). These two parameters are evaluated at current time step n using the updated temperatures at the nodes. Furthermore, the degree of cure dependent chemical shrinkage and thermal expansion (with its additional dependence on temperature), see Equation (3.66), can be expressed in the discretized form as

$$\begin{aligned} g\left(\alpha_{pn}\right) &= 1 + \beta_c \alpha_{pn} \\ \phi\left(\alpha_{pn}, \Theta_{pn}\right) &= 1 + \left[(1 - \alpha_{pn})\beta_{\Theta_f} + \alpha_{pn}\beta_{\Theta_s}\right](\Theta_{pn} - \Theta_0) \end{aligned} \tag{4.12}$$

The thermal effects in the SLA simulation are also modeled with respect to the OTM framework and its meshfree discretization. Considering the heat conduction equation (3.91), its discretized form can be written as

$$\text{Div Q} = - \int_{\Omega} \kappa \, grad \, \delta\Theta \, grad\Theta \, dv = - \sum_{p}^{n_{mp}^I} \sum_{p}^{n_{mp}^I} \boldsymbol{B}_I(\boldsymbol{x}_{pn})\boldsymbol{B}_I(\boldsymbol{x}_{pn})\kappa_{pn}\Theta_{In}v_{pn} \tag{4.13}$$

The absorbed laser power which is defined at a material point can be written in the discretized form at a node as

$$
\mathrm{E}_{Laser,In} = \sum_{p}^{n_{mp}^I} N_I(\boldsymbol{x}_{pn}) \mathrm{E}_{Laser,pn} \tag{4.14}
$$

where $\mathrm{E}_{Laser,pn}$ can be computed according to Equation (2.2). The thermo-chemical specific heat capacity in Equation (3.63) can be represented as

$$
c_{\Theta C} = \sum_{p}^{n_{mp}^I} \left[\left(a_F \Theta_{In} + b_F \Theta_{In}^2 \right)(1 - \alpha) + \left(a_S \Theta_{In} + b_S \Theta_{In}^2 \right) \alpha \right] N_{In}(\boldsymbol{x}_{pn}) m_{pn} \tag{4.15}
$$

Besides the equation of motion, the final derived energy equation (3.93) is obtained in the discretized form within the OTM framework as

$$
\begin{aligned}
c_{\Theta C}\dot{\Theta}_{In} ={}& \left(\frac{\varphi J_M \operatorname{tr}(\boldsymbol{\sigma})}{3\rho_0} \frac{\partial g}{\partial \alpha} - \frac{\partial \Psi_{\Theta C}}{\partial \Theta} - \Theta_{In} \frac{J_M \operatorname{tr}(\boldsymbol{\sigma})}{3\rho_0} \frac{\partial \varphi}{\partial \Theta} \frac{\partial g}{\partial \alpha} + \frac{\partial^2 \Psi_{\Theta C}}{\partial \Theta^2} \right) \dot{\alpha} \\
& - \Theta_{In} \left(\frac{g J_M \operatorname{tr}(\dot{\boldsymbol{\sigma}})}{3\rho_0} \frac{\partial \varphi}{\partial \Theta} \right) - \sum_{p}^{n_{mp}^I} \sum_{p}^{n_{mp}^I} \boldsymbol{B}_I(\boldsymbol{x}_{pn}) \boldsymbol{B}_I(\boldsymbol{x}_{pn}) \kappa_{pn} \Theta_{In} v_{pn} \\
& + \sum_{p}^{n_{mp}^I} N_I(\boldsymbol{x}_{pn}) \mathrm{E}_{Laser,pn}
\end{aligned} \tag{4.16}
$$

4.3 Time Integration

Suitable time integration scheme is used to solve (4.9). In this work, the central difference time integration scheme have been used. This time integration scheme can be derived from the explicit Newmark scheme, see (HUANG ET AL., 2019) for more details.

Explicit Newmark scheme

Within the Newmark scheme, the variables at t_{n+1} can be described interms of variables at time t_n as

$$
\boldsymbol{u}_{n+1} = \boldsymbol{u}_n + \Delta t \boldsymbol{v}_n + \left(\frac{1}{2} - \beta \right) \Delta t^2 \boldsymbol{a}_n + \beta \Delta t^2 \boldsymbol{a}_{n+1} \tag{4.17}
$$

$$
\boldsymbol{v}_{n+1} = \boldsymbol{v}_n + (1 - \gamma) \Delta t \boldsymbol{a}_n + \gamma \Delta t \boldsymbol{a}_{n+1} \tag{4.18}
$$

where β and γ are the Newmark scheme parameters and the explicit Newmark scheme is obtained when β is set to 0.

Central Difference Scheme

The central difference scheme can be derived from the Newmark scheme by further setting the parameter $\gamma = 0.5$. Displacements at time step t_n can be written as

$$\boldsymbol{u}_n = \boldsymbol{u}_{n-1} + \Delta t \boldsymbol{v}_{n-1} + \frac{1}{2}\Delta t^2 \boldsymbol{a}_{n-1} \tag{4.19}$$

and the velocity at time step t_n can be expressed as

$$\boldsymbol{v}_n = \boldsymbol{v}_{n-1} + \frac{(\boldsymbol{a}_n + \boldsymbol{a}_{n-1})\,\Delta t}{2}. \tag{4.20}$$

(4.19) can be rewritten as

$$\boldsymbol{v}_{n-1} + \frac{\Delta t}{2}\boldsymbol{a}_{n-1} = \frac{1}{\Delta t}\left(\boldsymbol{u}_n - \boldsymbol{u}_{n-1}\right). \tag{4.21}$$

(4.20) can be written in terms of (4.21) as

$$\Delta t \boldsymbol{v}_n = \boldsymbol{u}_n - \boldsymbol{u}_{n-1} + \frac{\Delta t^2}{2}\boldsymbol{a}_n \tag{4.22}$$

The velocity at time t_n can be represented in terms (4.22) and (4.19) as

$$\boldsymbol{v}_n = \frac{\boldsymbol{u}_{n+1} - \boldsymbol{u}_{n-1}}{2\Delta t} \tag{4.23}$$

Here, the velocity $\boldsymbol{v}_n$ has been expressed in terms of displacements at time steps t_{n+1} and t_{n-1}. Subsequently, the acceleration at time t_n can be obtained as

$$\boldsymbol{a}_n = \frac{(\boldsymbol{u}_{n+1} - \boldsymbol{u}_n) - (\boldsymbol{u}_n - \boldsymbol{u}_{n-1})}{\Delta t^2}. \tag{4.24}$$

Time Integration within OTM Method

By applying the central difference scheme, the displacements at time t_{n+1} can be obtained by inserting the acceleration $\boldsymbol{a}_n$ in (4.24) to Equation (4.8) at time t_n. Subsequently, using the obtained acceleration $\boldsymbol{a}_n$, the displacements at time t_{n+1} can be defined by using the acceleration $\boldsymbol{a}_n$. The Equation (4.24) can be rewritten to obtain the displacement at time t_{n+1}

$$\boldsymbol{u}_{n+1} = 2\boldsymbol{u}_n - \boldsymbol{u}_{n-1} + \Delta t^2 \boldsymbol{a}_n \tag{4.25}$$

Special initialization is required for $n = 0$ in (4.25) and using second order accurate Taylor series expansion, the displacement at $\boldsymbol{u}_{-1}$ can be defined as

$$\boldsymbol{u}_{-1} = \boldsymbol{u}_0 - \Delta t \boldsymbol{v}_0 + \frac{\Delta t^2}{2}\boldsymbol{a}_0 \tag{4.26}$$

Current velocity at time t_{n+1} has been defined to efficiently implement the central difference time integration scheme in (4.25)

$$\hat{\boldsymbol{v}}_{n+1} \;=\; \boldsymbol{v}_n + \frac{\Delta t}{2}\boldsymbol{a}_n, \; n = 0 \tag{4.27}$$

$$\hat{\boldsymbol{v}}_{n+1} \;=\; \hat{\boldsymbol{v}}_n + \Delta t \boldsymbol{a}_n, \; n \geq 1 \tag{4.28}$$

Accordingly, displacement at time t_{n+1} is updated as

$$\boldsymbol{u}_{n+1} \;=\; \boldsymbol{u}_n + \Delta t \hat{\boldsymbol{v}}_{n+1} \tag{4.29}$$

A time discretisation is required for the energy equation (4.16). Applying Euler forward scheme for this first order differential equation, the temperature at the node I is updated as

$$\Theta_{I\,n+1} \;=\; \Theta_{I\,n} + \Delta t \dot{\Theta}_{I\,n+1} \tag{4.30}$$

Similarly, the degree of cure at the material point is updated as

$$\alpha_{p\,n+1} \;=\; \alpha_{p\,n} + \Delta t \dot{\alpha}_{p\,n+1} \tag{4.31}$$

where $\dot{\alpha}_{p\,n+1}$ is obtained from the cure reaction rate (2.5).

4.4   Update of Primary Variables

Now, the approximation equation (4.8) can be transformed into an algebraic equation by decomposing into a set of independent nodal equilibrium equations. Using the acceleration at time t_n in Equation (4.24) and the concept of lumped mass matrix, the equilibrium of the body is transformed into a set of independent nodal equilibrium equations. This step is equivalent to the Finite Element Method framework given in (BATHE, 2006), for instance

$$m_{I\,n} \frac{(\boldsymbol{u}_{n+1} - \boldsymbol{u}_n) - (\boldsymbol{u}_n - \boldsymbol{u}_{n-1})}{\Delta t^2} \;=\; \boldsymbol{p}_{I\,n} + \boldsymbol{r}_{I\,n}. \tag{4.32}$$

Within the Galerkin meshfree framework, the boundary forces, $\boldsymbol{p}_{I\,n}$, are prescribed only at the nodes of the Neumann boundary. In order to compute the acceleration efficiently, lumped mass matrix was applied to the equilibrium equation (4.32). The full mass matrix becomes the diagonal matrix by the application of row-sum technique and leading to the direct computation of the nodal mass, which can be represented as

$$m_{I\,n} \;=\; \sum_{p}^{n_{mp}^{I}} N_{I\,n}(\boldsymbol{x}_{p\,n}) m_p \tag{4.33}$$

where $m_{I\,n}$ is the mass at node I. The nodal residual vector (or internal forces) $\boldsymbol{r}_{I\,n}$ in Equation (4.32) can be formulated as

$$\boldsymbol{r}_{I\,n} \;=\; \sum_{p}^{n_{mp}^{I}} \left[N_{I\,n}(\boldsymbol{x}_{p\,n}) \hat{\boldsymbol{b}}_{p\,n} m_p \;-\; \boldsymbol{B}_I(\boldsymbol{x}_{p\,n}) \boldsymbol{\sigma}_{p\,n} v_p \right] \tag{4.34}$$

Algorithm 1 Algorithmic implementation of a time step in OTM

Require: Initial nodal set and material point set

1. Compute local mass matrix and local nodal force vector
2. Update primary variables and nodal coordinates
3. Update material point coordinates
4. Constitutive updates at material point
5. Search Algorithm to update support domains
6. Recompute shape functions
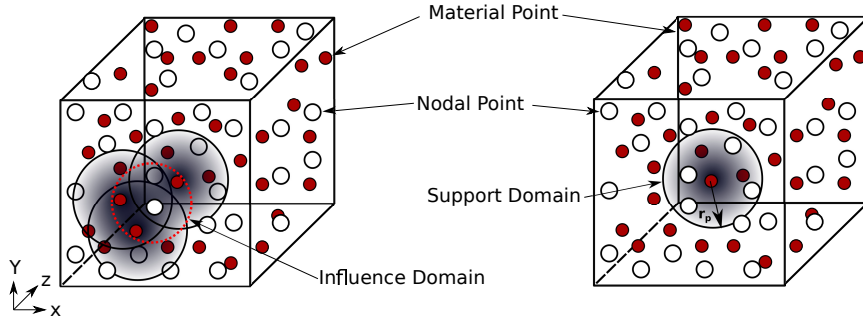
Figure 4.1. Influence Domain of a node and Support Domain of a material point

where n_{mp}^I is the number of material points in the influence domain of Node I. The corresponding material points within each influence domain can be determined from the support domain directly, without any need of additional search algorithm, see Figure 4.1.

In most of the cases, due to presence of more than 4 nodes within each support domain, there is nonlinear distribution of displacements leading to distortion, see FLANAGAN & BELYTSCHKO (1981). A stabilization term is added to the nodal residual vector to penalize the inaccurate behavior due to underintegration within every support domain, see (WEISSEN-FELS & WRIGGERS, 2018):

$$\boldsymbol{r}_{In-stab} = \boldsymbol{r}_{In} - \varepsilon \sum_p^{n_{mp}^I} N_{In}(\boldsymbol{x}_{pn})\boldsymbol{e}_{I,pn} \qquad (4.35)$$

where ε is the penalty parameter and $\boldsymbol{e}_{I,pn}$ is the error due to underintegration. The second term on the right side of Equation (4.35) enforces error to be zero by penalty regularization. Within the OTM method, the error due to underintegration, $\boldsymbol{e}_{I,pn}$, is due to the nonlinear distribution of displacements and calculated based on the position of nodal point I and material point p and the error is determined by

$$\boldsymbol{e}_{I,pn} = \frac{\boldsymbol{x}_{In} - \boldsymbol{x}_{pn} - (\tilde{\boldsymbol{x}}_{In} - \tilde{\boldsymbol{x}}_{pn})}{\|\boldsymbol{x}_{In-1} - \boldsymbol{x}_{pn-1}\|}, \qquad (4.36)$$

where, $\boldsymbol{x}_{In} - \boldsymbol{x}_{pn}$ is the distance vector between material point p and nodal point I, $\tilde{\boldsymbol{x}}_{In} - \tilde{\boldsymbol{x}}_{pn}$ is the distance vector updated from the constant increment of deformation gradient

$$\tilde{\boldsymbol{x}}_{In} - \tilde{\boldsymbol{x}}_{pn} = \Delta\boldsymbol{F}_{pn}[\boldsymbol{x}_{In-1} - \boldsymbol{x}_{pn-1}], \qquad (4.37)$$

where $\Delta \boldsymbol{F}_{pn}$ is the increment of the deformation gradient, as computed in (4.40). The error due to underintegration in Equation (4.36) is characterzed by the difference of the distance vectors between two formulations. If the distribution of displacements in the support domain is linear, the difference vectors in Equation (4.36) is same as computed from the OTM method. By solving (4.32), the nodal position vector can be updated from the displacement increments of the next computation step

$$\boldsymbol{x}_{I\,n+1} = \boldsymbol{x}_{I\,n} + \Delta \boldsymbol{u}_{I\,n+1} \tag{4.38}$$

where $\boldsymbol{x}_{I\,n}$ is the initial nodal coordinates.

4.5 Update of Kinematic Quantities

The position of a material point at the next time step is updated by multiplying the shape functions at current time step with the nodal coordinates of the next time step

$$\boldsymbol{x}_{p\,n+1} = \sum_{I}^{n_{np}^{n}} N_{I\,n}(\boldsymbol{x}_{pn}) \boldsymbol{x}_{I\,n+1} \tag{4.39}$$

where n_{np}^{n} is the number of support nodes of material point p. During large deformations, the support nodes of a material point are changing at every time step. Within the nonlinear regime, strong movement of the particles can lead to large deformations of the body, making the deformation gradient an important kinematic quantity. Using the updated Lagrangian formulation within the OTM framework, the deformation gradient at the next time step, $n+1$,

$$\boldsymbol{F}_{p\,n+1} = \Delta \boldsymbol{F}_{p\,n+1} \boldsymbol{F}_{p\,n} \tag{4.40}$$

is updated in terms of the current value of the deformation gradient, $\mathbf{F}_{p\,n}$, and the increment of the deformation gradient is

$$\Delta \boldsymbol{F}_{p\,n+1} = \mathbf{1} + \sum_{I}^{n_{np}^{n}} \frac{\partial N_{I\,n}(\boldsymbol{x}_{pn})}{\partial \boldsymbol{x}} \Delta \boldsymbol{u}_{I\,n+1} \,, \tag{4.41}$$

Accordingly, the volume and mass at each material point can be updated based on $\Delta \boldsymbol{F}_{p\,n+1}$

$$v_{p\,n+1} = \det(\Delta \boldsymbol{F}_{p\,n+1}) v_{p\,n} \tag{4.42}$$

$$\rho^{n+1} = \frac{m_p}{v_{p\,n+1}} \tag{4.43}$$

For the SLA simulation, the thermo-chemo-mechanical split of the deformation gradient (3.64) can be updated as

$$\boldsymbol{F}_{p\,n} = \boldsymbol{F}_{M,pn} \boldsymbol{F}_{C,pn} \boldsymbol{F}_{\Theta,pn} \tag{4.44}$$

where $\boldsymbol{F}_{C,pn}$ and $\boldsymbol{F}_{\Theta,pn}$ are dependent on scalar functions of chemical shrinkage and thermal expansion respectively, see Equations (3.65) and (4.12). Accordingly, the mechanical part of the deformation gradient can be expressed as

$$\boldsymbol{F}_{M,pn} = \boldsymbol{F}_{pn}\boldsymbol{F}_{\Theta,pn}^{-1}\boldsymbol{F}_{C,pn}^{-1} \tag{4.45}$$

and it can be used to compute the mechanical right Cauchy-Green tensor (3.72) at the material point as

$$\boldsymbol{C}_{M,pn} = \boldsymbol{F}_{M,pn}^{T}\boldsymbol{F}_{M,pn} \tag{4.46}$$

Similarly, the degree of cure dependent shear modulus of the equilibrium spring (3.67) can be written as

$$\mu_{\alpha}(\alpha_{pn}) = p_{\alpha_1}^{\mu}(1 - \alpha_{pn}) + p_{\alpha_2}^{\mu}\alpha_{pn} \tag{4.47}$$

which is used to compute the cure dependent relaxation time $\tau(\alpha_{pn})$. Now, using the evolution equation (3.68), the elastic-inelastic split of viscoelastic deformation gradient within the Maxwell branch is updated as

$$\boldsymbol{C}_{M_{in_i},pn} = \boldsymbol{C}_{M_{in_i},pn-1} + \frac{\Delta t}{\tau(\alpha_{pn})}\left(\boldsymbol{C}_{M,pn} - \boldsymbol{C}_{M_{in_i},pn-1}\right) \tag{4.48}$$

Finally, the discretized mechanical second Piola-Kirchoff stress tensor (3.86) is defined as

$$\begin{aligned}
\boldsymbol{S}_{M,pn} = \frac{2}{g\left(\alpha_{pn}\right)\varphi\left(\alpha_{pn},\Theta_{pn}\right)}&\left[J_M^{-\frac{2}{3}}\mu_{\infty,pn}\left(\mathbf{I} - \frac{1}{3}\boldsymbol{C}_{M,pn}^{-1}\operatorname{tr}(\boldsymbol{C}_{M,pn})\right) + \right.\\
\frac{K}{2}\left(J_M^2 - 1\right)\boldsymbol{C}_{M,pn}^{-1} &+ J_M^{-\frac{2}{3}}\mu_{i,pn}\left.\left(\boldsymbol{C}_{M_{in_i},pn}^{-1} - \frac{1}{3}\boldsymbol{C}_{M,pn}^{-1}\operatorname{tr}(\boldsymbol{C}_{M,pn}\boldsymbol{C}_{M_{in_i},pn}^{-1})\right)\right]
\end{aligned} \tag{4.49}$$

A summary of the evaluated quantities in this chapter have been presented at the end of Chapter 3.

4.6 Local Max- Ent shape functions

In meshfree methods, the polynomial basis functions, which are normally used within the finite element framework, are not appropriate. In OTM method, local maximum entropy (LME) (ARROYO & ORTIZ, 2006) approximation function is used which has to be determined for an arbitrary number of nodes within the support domain. The LME shape functions possess weak Kronecker-δ property at the boundary and it is fulfilled only on convex boundaries, see (LI ET AL., 2010). Also, the LME shape functions does not fulfill either the first order completeness or the partition of unity condition. In order to achieve convergence to the correct solution of the equation of motion, computational algorithms should fulfill these basic conditions, see (HUGHES, 1987; BELYTSCHKO ET AL., 1998).

For a given material point, the LME shape function can be derived by solving a constrained

optimization problem

$$
\begin{aligned}
\text{Minimum} : f_\beta \left[N_I(\boldsymbol{x}_p) \right] &= \beta U \left[N_I(\boldsymbol{x}_p) \right] - H \left[N_I(\boldsymbol{x}_p) \right] && (4.50) \\
\text{Subject to} : N_I(\boldsymbol{x}_p) &\geq 0, && (4.51) \\
\sum_{I}^{n_{np}} N_I(\boldsymbol{x}_p) &= 1, && (4.52) \\
\sum_{I}^{n_{np}} N_I(\boldsymbol{x}_p)\boldsymbol{x}_1 &= \boldsymbol{x}_p && (4.53)
\end{aligned}
$$

where β parameter controls the degree of locality. The function $U \left[N_I(\boldsymbol{x}_p) \right]$ describes the locality as

$$
U \left[N_I(\boldsymbol{x}_p) \right] = \sum_{I}^{n_{np}} N_I(\boldsymbol{x}_p) \left| \boldsymbol{x}_p - \boldsymbol{x}_I \right|^2, \tag{4.54}
$$

and the function $H \left[N_I(\boldsymbol{x}_p) \right]$ describes the entropy as

$$
H \left[N_I(\boldsymbol{x}_p) \right] = - \sum_{I}^{n_{np}} N_I(\boldsymbol{x}_p) \log N_I(\boldsymbol{x}_p). \tag{4.55}
$$

The LME shape functions has an exponential ansatz and belongs to the class of radial basis functions. By enforcing first order completeness condition (4.53) using the Lagrangian multiplier method and the partition of unity condition (4.52) through normalization, the unique solution of $N_I(\boldsymbol{x}_p)$ for this optimization problem can be obtained as, see (ARROYO & ORTIZ, 2006),

$$
N_I(\boldsymbol{x}_p) = \frac{Z_I(\boldsymbol{x}_p)}{Z}, \; Z_I = \exp^{-\beta |\boldsymbol{x}_p - \boldsymbol{x}_I|^2 + \lambda(\boldsymbol{x}_p - \boldsymbol{x}_I)}, \; Z = \sum_{I}^{n_{np}} Z_I(\boldsymbol{x}_p) \tag{4.56}
$$

where λ is a Lagrangian multiplier, which is determined by solving $\sum N_I(\mathbf{x}_p)(\mathbf{x}_p - \mathbf{x}_I) = 0$ using Newton- Raphson algorithm. The parameter β is calculated as $\beta = \frac{\gamma}{h^2}$, where γ controls the degree of locality of LME shape functions and it should be in the range of 0.8 to 4, and h is the characteristic nodal spacing.

The main goal of the OTM method in this work is development of a parallel framework for its solution and its applications to large deformation problems and Stereolithography process. In the next Chapter, a detailed discussion is presented regarding the development of parallel framework for the OTM method and the scalability performance of this multiprocessing approach for the numerical solutions of two large deformation problems is investigated.

Chapter 5

Parallel Architecture with MPI

Development of MPI framework requires carefully thought strategies for data-partitioning and data communication. The parallel framework utilizes the MPI for communication and synchronization between processes. MPI is a standard paradigm for implementing parallel framework in distributed memory platforms, (BALAJI ET AL., 2010; PLIMPTON & DEVINE, 2011). In order to exchange message and manage processes, MPI provides a collective set of library routines. It is generally used in high end computing applications involving intensive calculations (NOTAY & NAPOV, 2015). Data-partitioning refers to the process of dividing the problem domain into smaller subdomains. Subdomain geometry affects the scalability since it is associated with equivalent work load through load-balancing. In order to achieve high scalability, subdomains are expected to contain same amount of work (load balancing) while minimizing the need for communications. Several approaches exist in field of non-overlapping domain decomposition methods to solve the challenging mechanical heterogeneous problems on massively parallel architectures. A new parallel mesh generation method has been developed by (GHARBI ET AL., 2021) which leads to subdomains with shape well-suited for Schur based domain decomposition methods, such as FETI (FARHAT & ROUX, 1991) and BDD (MANDEL, 1993) solvers. Another domain decomposition method, Orthogonal Recursive Bisection (ORB) algorithm has been implemented by (OGER ET AL., 2016; YANG ET AL., 2020) which has been shown to lead to scalable results at large processor numbers. Also, due to the distributed nature of the framework, duplication of the data is required for communication, resulting in the increased overall memory requirement.

Generally when a parallel program is run on several processes simultaneously, there are data dependencies between the tasks. A process might need intermediate results in order to carry out its computations and this intermediate result could be located on adifferent process. Hence, bottlenecks occur which slow down the computation. The MPI library provides different communication primitives: point-to-point and collective communication. One way to maximize the performance of parallelization is to reduce the overheads due to communication operations. Overhead is defined as *the length of time that a processor is engaged*

in the transmission or reception of each message; during this time, the processor cannot perform other operations (CULLER ET AL., 1993). Collective communication operations, introduced in the latter versions of MPI, have been a key concept used in large scale parallel applications to minimize the communication overheads (BARIGOU & GABRIEL, 2017; BARIGOU ET AL., 2015). Although they are widely used due to their increased productivity and performance, there are some limitations. Due to the dependencies on all the processes of a communicator, there exist scalability issues and conventional collectives support limited communication patterns, such as, broadcast and all-to-all, see (GHAZIMIRSAEED ET AL., 2020). In order to address these issues, Neighborhood Collectives, introduced by the MPI 3.0, provide an alternative to the users to define arbitrary communication patterns. This can be used to implement nearest neighbor collective operations where each process interacts with only a small neighborhood of processes. Neighborhoods can be described either by Cartesian neighborhoods or by general communication graphs, for more details see (HOEFLER ET AL., 2011; MPI FORUM, 2015).

When performing computations within each subdomain, all the necessary information should be available in the same process. But, for the subdomain's boundary, some of the required information will be located on other processes. Hence, a communication pattern is required, the most commonly used is halo regions. At every computation step, the halo regions are exchanged with the neighboring processes so that every process can access to the necessary information. The goal of the halo regions is to locally replicate the domain residing in other processes. At every computation step, when processes communicate with their neighbors, performance of any process depends strongly on the performance of its neighbors. This can result in delays (LAOIDE-KEMP, 2015). Additionally, the probability for delays can increase with the number of processes.

Large scale computations requires an adaptive code to run efficiently on distributed memory systems. Good data management and domain decomposition are critical parameters. Some of the cumbersome tasks during simulation using the OTM method in a parallel environment involves modifying, deleting and adding particles in a subdomain, adjusting the subdomain partitioning dynamically and performing migration of particles to maintain load balance during the simulations. These tasks require flexible and efficient data management scheme. In mesh-based methods, flexible and efficient data management schemes for parallel systems have been implemented for adaptive hp finite element method (LASZLOFFY ET AL., 2000), and for simulation tool for geophysical mass flows (PATRA ET AL., 2005). In meshfree methods, (CAO ET AL., 2017) developed data management strategies for a MPI parallel implementation of the SPH method to simulate volcano plumes. (FERRARI ET AL., 2009) used a flexible way in linked lists using pointers so that particles can be deleted or added during the simulation. Similar approach for modifying pointer-based information has been adopted in the current work.

The following three reasons motivate the work presented in this chapter. First, while parallelization approach to OTM method has been implemented by (LI ET AL., 2014), efficient implementation strategies have been presented by introducing communication for both nodal and material point halo regions for localized updates within every subdomain. Second, with the use of improved data structures for halo regions, flexibility have been introduced to han-

dle variable workloads (dynamic halo regions). This is helpful when new nodal or material point quantities are added into the communication. Finally, with the use of nearest neighborhood communication for neighbor detection and communication, the communication costs are reduced even though total halo particles increases with increase in number of subdomains.

In this work, computational strategies are proposed for parallel processing of OTM Method using MPI for scalability on large-scale computer clusters. Mechanisms are presented for efficient addition or removal of nodes and material points from their corresponding influence and support domains respectively, thereby reducing computational overheads. Hence, storage issues related to the fixed-size arrays are eliminated. Dynamic halo region is implemented that can handle variable workloads. Both nodes and material points within every subdomain and their corresponding influence and support domains are managed by STL map which can ensure quick and flexible access and modifications. In order to ensure good static and dynamic load balance, Recursive Coordinate Bisection (RCB) algorithm, a Cartesian based decomposition method is used for both static and dynamic decomposition (dynamic load balancing). The RCB decomposition method facilitates good scalability by ensuring minimum interfacial surface area between the sub-domains. Parallel decomposition of spatial domain is carried out in such a way that each subdomain is physically compact and the computations can be performed locally at each process. The flexibility of our data access methodology, data structures, dynamic halo regions enables efficient implementation of OTM method within parallel framework. To reduce global communications, nearest neighbor communication operations are implemented using MPI collectives.

5.1 Software Design

The parallel framework is written for use on multi-CPU architectures. The parallel codes are written in C++ and make use of its object-oriented features. The approach to parallelize the OTM method with MPI is to separate the spatial domain into distinct subdomains and allocate nodes and material points to each MPI process, such that each process treats its own subdomain independently. One advantage of this approach is the minimum impact on the contents and structure of a serial code. Halo regions of nodes and material points are then distributed between the subdomains at every computation step such that the primary nodal variables and constitutive updates at the material points can be computed in parallel.

5.1.1 Domain Decomposition

To decompose the domain, the Recursive Coordinate Bisection (RCB) algorithm is used. The objective of the partitioning library is to provide a initial computational workload which is uniformly distributed. This is accomplished by a distribution of almost equal number of particles (nodes and material points) in each process. Domain decomposition is conducted by cutting along the partition planes in the spatial domain recursively (Figure 5.1). Each sub-domain is assigned to one process. Hence, the decomposition depends on the number of processes and the domain size (SELVAM & HOFFMANN, 2015). The goal of using this

domain decomposition algorithm is to ensures geometrical locality of the particles and to simplify the creation of halo regions. Both nodes and material points carry their influence and support domain information respectively during the distribution process.
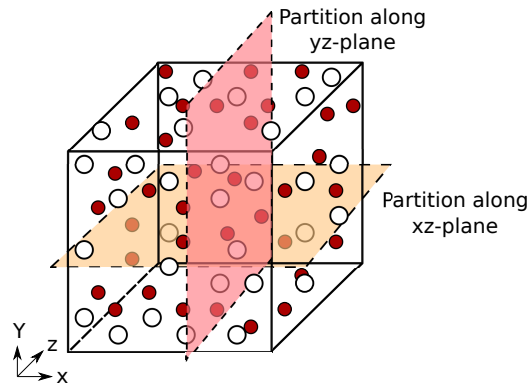

Figure 5.1. Domain decomposition into four processes using RCB.

Movement of particles and subsequent adjustments in the subdomain will cause load-imbalance among the processes. The computational load at a given time interval is monitored to re-assign the workload evenly among the processes and to minimize the communication. For the purpose of dynamic load balancing, the Recursive Coordinate Bisection algorithm is called at optimized time intervals, within the mid-increment of the time step, to update the subdomain boundaries if required.

5.1.2   Dynamic Halo Regions

The nodal and material point updates are performed in each subdomain in parallel. For a node and material point, its influence and support domain could be spread across multiple processes (Figure 5.2). Nodes and material points, which are close to the division boundaries of subdomains need to share information. For this, the halo regions are necessary. These halo regions are copies of nodal and material point data that are sent to neighbor processes via two communication steps.
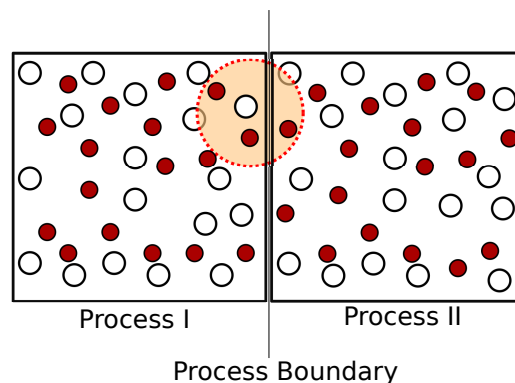

Figure 5.2. Influence Domain of a node spread across multiple processes
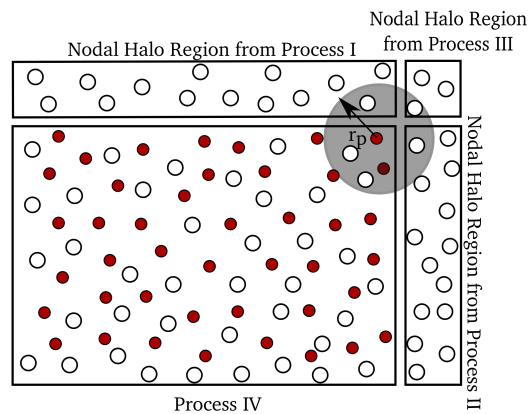
Figure 5.3. Nodal halo region and support domain with support nodes from halo region.

After nodal and material point updates, the halo regions are constructed dynamically during the computation, depending on the amount of communication. The first round involves nodal halo communication for the material point updates, where position, velocity, influence domain and other nodal data are communicated. Afterward, all data at material points can be computed within each subdomain. For material points whose support nodes are located in neighboring subdomains, its support domain is reconstructed through halo nodes (Figure 5.3). Hence, support domains are formed using nodes in its own subdomain and nodal halo region. After the update of material point information, the second round of communication involves material point halo communication for the nodal updates. Similarly, influence domain of nodes at the boundary of subdomains are reconstructed through material point halo regions (Figure 5.4). Nodal updates take place locally at each subdomain using the information from its own subdomain and from material point halo region.
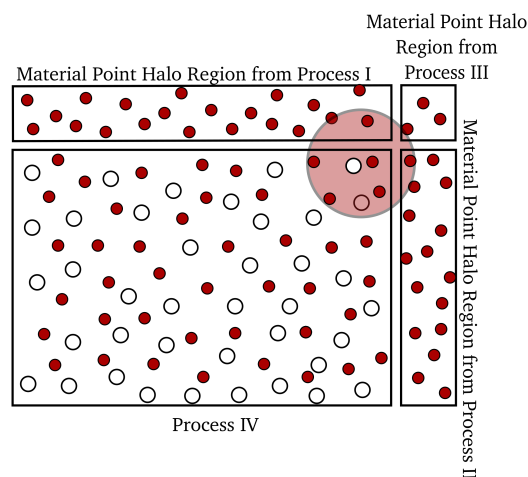

Figure 5.4. Material point halo region and influence domain using material points from halo region.

At the beginning of the halo communication process, the neighbors of each subdomain need to be detected. This will enable every subdomain to initiate communication locally only

with its neighbors and to implement the nearest neighbor communication patterns (*sparse collective operations*). In general, for large scale applications, efficient implementation of sparse collective communication operations is most important (HOEFLER & TRÄFF, 2009). At every time step, identification of nearest neighbors of a subdomain is facilitated through the process of bounding box intersection with its neighbor subdomains. A set of local neighborhoods (*process neighborhood*) is defined for every subdomain (Figure 5.5). Each process neighborhood consists a list of k target processes and a list of k source processes. For each subdomain, halo regions will be sent to target processes and simultaneously, it will receive halo regions from the same target processes. So, the source and target processes are same for each process but the amount of information to be received and sent may differ. Bounding box consists of coordinate information of both nodes and material points located at the lower and upper bounds of each subdomain and it is recomputed at every computation step. In Figure 5.6, $(B_{max}^{II}, B_{min}^{II})$ represents the bounding box of Process II and $(B_{max}^{I}, B_{min}^{I})$ represents the bounding box of Process I. Before performing intersection, bounding boxes from neighboring processes are extended by a width equivalent to maximum support radius of the sub-domain. This is performed at every time step to determine the extent of overlap of bounding boxes, i.e. halo regions. The maximum support radius at each sub-domain is used to extend the bounding boxes gathered from neighboring processes. Width of halo region for each sub-domain is identified as the overlap region between the bounding boxes of each sub-domain.
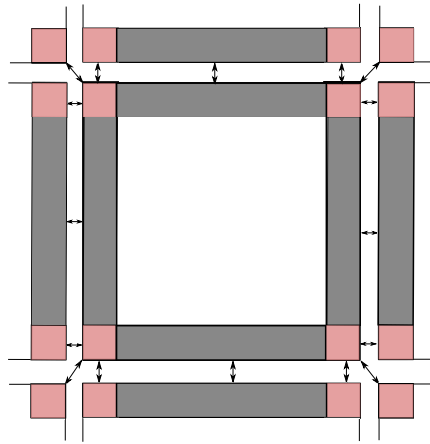

Figure 5.5. Schematic Representation of Nearest Neighborhood Communication for a typical halo region update operation, showing the halo exchanges for the faces and corners of sub-domains: Grey regions represent the halo communication among the faces of the sub-domains (each with one neighbor) and red regions represent the halo exchange among the corner parts of the sub-domains (each with three neighbors).

Nodes within every intersecting bounding boxes are identified as halo nodes, see Figure 5.6. Halo nodes which are to be sent, are serialized, i.e., the nodal data is represented as a large array of chars and stored in a buffer. Halo nodes sharing boundary with each neighbor is sent to the specific neighboring sub-domain (Figure 5.7). Here, MPI virtual topology functionality is used for sparse collective operations, it uses the set of local neighborhoods,
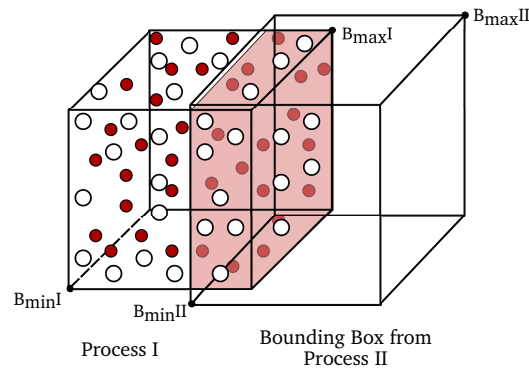
Figure 5.6. Bounding Box intersection between Process I and Process II and Identification of halo nodes in Process I in the intersection region is depicted in the Figure. Nodes in the intersection region will be sent as halo nodes to Process II.

i.e., source and target lists. Graph topology interface is used as it provides full flexibility in describing neighborhoods and the communication graphs are not limited to symmetric exchange patterns, which is in contrast to the Cartesian topology mechanism (FORUM, 2012). Before sending the halo nodes, the processes, at first, communicate how many nodes are to be exchanged along with total size of nodal data. After determining the total size of each buffer, memory allocation is made in the target process where the halo region is to be received (receive buffer) from its neighbor (Figure 5.7). The overall nodal halo communication step is sketched in Algorithm 2.

Algorithm 2 Nodal Halo Communication Step

Require: Bounding box computation at every process
Require: Detection of neighbor processes
   1. Exchange bounding box with all processes using MPI_Allgather
   2. Identify intersecting bounding boxes
   3. Identify nodes at the intersection (or Halo nodes) to be sent
Require: Create local process neighborhood using MPI_Dist_graph_create_adjacent
   1. Determine the nodal size for halo.
   2. Exchange the nodal sizes with nearest neighbors using MPI_Neighbor_alltoallv
   3. After receiving the nodal sizes at destination process, allocate memory for receive buffer
   4. Pack the halo nodes to be sent in a send buffer.
   5. Exchange the halo node information using MPI_Neighbor_alltoallv.

After the nodal halo communication, material points perform the search process using nodal information from its own subdomain and from nodal halo region. In order to improve the computational efficiency of search algorithms, the linked cell method (GRIEBEL ET AL., 2007) has been implemented. Linked cell method within OTM method is a new feature in this work. In solids undergoing large deformations or in fluid simulations, both the nodes and material points may change its position with time. An efficient search algorithm is needed to dynamically update the support domain while solving the equations without incurring
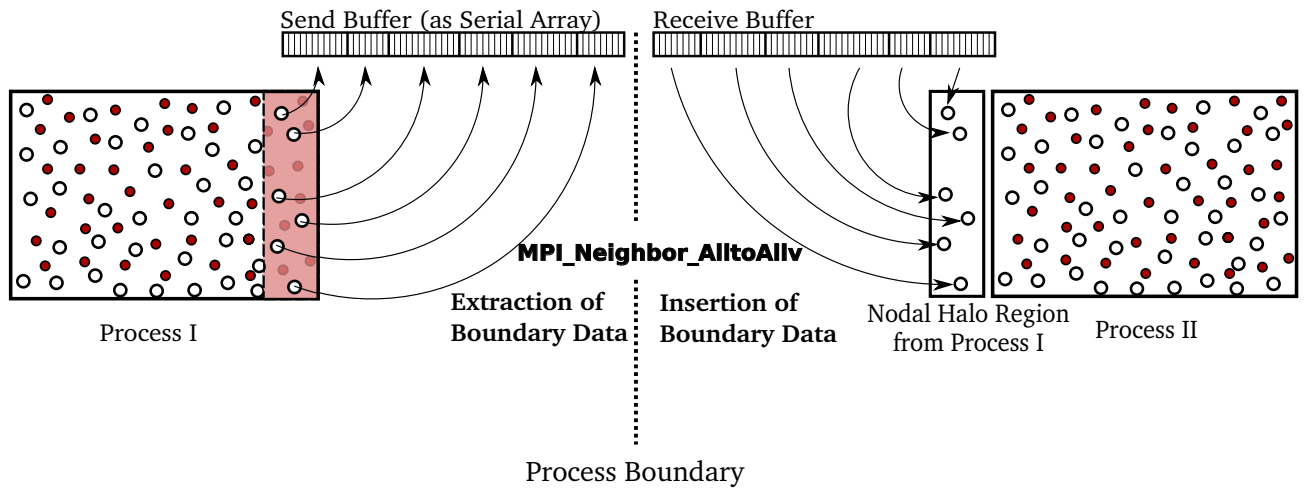
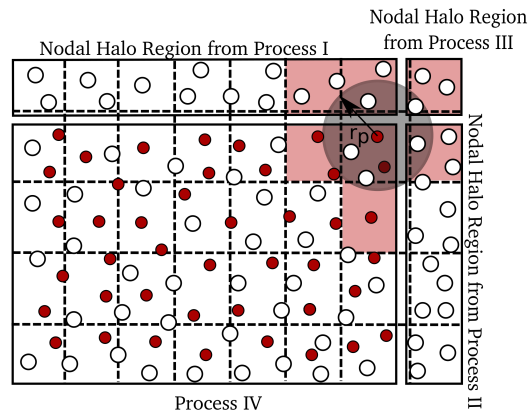**Figure 5.7.** MPI Communication from Process I to Process II

Figure 5.8. Linked Cell Method

excessive computational costs. Linked cell method significantly reduces the computational efforts, when the number of particles is large. The main idea of the linked list is to map the nodal positions on a grid. All the nodes have a unique particle ID and a data structure stores all the information of each grid. For each cell, a list of nodal IDs and pointer to those nodes are stored. After the nodal updates and subsequent formation of nodal halo regions, both the subdomain and nodal halo region is subdivided into static cells (Figure 5.8). Only nodes in the vicinity of a material point are checked during the search process. This is done by identifying the cells which intersect with the search radius and nodes in those intersecting cells are considered within the search process. A brief sketch of the steps involved is shown in Algorithm 3.

After material point updates, communication of material point halo region follow in the same line as of nodal halo communication, starting with the bounding box intersection, communicating total number of material points to be exchanged, and communication of material point information.

Algorithm 3 Algorithmic scheme for Linked Cell Method

Require: Nodal Halo Region
Require: New support radius
    1. Divide the subdomain and nodal halo region into cells, see Figure 5.8
    2. Identify intersection of cells with the support domain
    3. Find the nodes which belong to the material point

5.1.3 Data Management Strategies

The core of parallel OTM method is formed by data structures and algorithms implemented as C++ templates. To store all the data of the nodes and material points, C++ classes have been defined. For the management of nodal and material point data (removal or addition), STL maps to store pointers to objects of nodal and material point data are preferred. This gives us flexibility for quicker removal and addition of nodal and material point data during load-balancing and for the formation of support and influence domains.

Information that is contained in a particle (node or material point) are its identifier (Global particle ID), coordinates, flags (indicators, such as, the particle is a node or material point and if the node is on the physical boundary of the problem domain) and its affiliate (rank of the process that the node or material point belongs to). Additional information that is contained in a node and material point is its pointer-based influence and support domain information respectively. With the help of this data structure, every subdomain handles pointers to objects of nodes and material points, bounding box information (maximum and minimum coordinates), and neighbor information (halo regions for nodes and material points). Choosing a proper way to handle this STL container depends on the problem itself. For instance, there is continuous update of support and influence domain in the OTM method, whose sizes can vary dynamically at every time step. After optimized intervals of dynamic load-balancing, the pointers to new particles (nodes and material points) are handled effectively by this container.

For nodal and material point halo communication, the data is packed into a serial array, whose size is varying. So, flexible data structures have been designed to pack all the information in the buffer. The message size for each node or material point is maintained as number of nodal or material point variables multiplied with the (size of double precision floating number), in order to prevent any kind of memory misalignment issues while packing information of mixed data types in a serial array.

The size of each nodal information is of arbitrary number of bytes due to the varying size of its influence domain. In (LI ET AL., 2014), MPI data structure was used to pack the nodal information. This restricts the information to be packed since only fixed-size information could be used to communicate. Here, the influence domain information of a node is packed more efficiently using a flexible size for every node. Similarly, for every halo material point, its support domain information is also included in the halo region. Packing support and influence information in halo region assists in localised updates within a subdomain. For instance, whenever the support domains of boundary material points are updated (Step 8 of Algorithm 4) and those material points are exchanged through halo communication, the updated sup-

---

Algorithm 4 Parallel OTM Time Step

For Process $\mathbb{P}^I$, $I = 1, ..., P$:

Require: Reading of Input information and Process $\mathbb{P}^I$ storing its own set of nodes and material points.

- Initial nodal set and material point set

- Initial support domain of material points

Require: Domain Decomposition by Zoltan, see Figure 5.1.

Require: Initial material point halo regions(steps are similar to Algorithm 2)
For computation step $t_k \rightarrow t_{k+1}$
1. Complete the influence domain with halo material points.
2. Compute the local mass matrix and local nodal force vector.
3. Update primary variables and nodal coordinates

Require: Nodal halo regions (for details, see Algorithm 2)

Require: Load balancing at optimized intervals, let's say at every time increments of t_{k+500}

- Clear both nodal and material point halo regions

- Call zoltan functions for load balancing (steps are similar to domain decomposition as in Section 5.1.1)

4. Complete the support domain with halo nodes, see Figure 5.3.
5. Update material point coordinates.
6. Constitutive updates at material point.
7. Division of subdomain and nodal halo region into cells (Linked Cell Method, see Figure 5.8)
8. Search algorithm to update the support domains
9. Recompute shape functions

Require: Material Point halo regions (steps are similar to Algorithm 2)

---

port domain information of halo material points will assist in updating the influence domain of the nodes locally at each sub-domain. This flexibility feature for packing any amount of information for halo communication is necessary for localized updates of nodes and material points.

Another advantage of using STL map for support and influence domain is that pointers to the support nodes or influence material points can be released while preserving their IDs. This proved to be helpful in situations where the support domains need to be constructed again using halo nodes after nodal updates. For instance, at time step t_k, support domains are updated (Step 8 of Algorithm 4). Subsequently, for the material point updates (Steps 4-7 of Algorithm 4) at time step t_{k+1}, the support domain computed at previous time step t_k will be used.

Object-oriented implementation, robustness and flexibility of the parallel framework to include additional physical phenomena are taken into consideration. With the use of *Eigen*

templated library, all the vector and matrix information are stored in contiguous memory locations and matrix operations are optimized.

5.2 Parallel Performance

The objective of parallelism is to perform simulation of larger and complex problems. To evaluate the ability of the parallel framework, strong scaling tests are conducted which measure the performance with increasing number of processes, keeping the problem size constant.

Every simulation is run for 2000 time steps. Variation of the computational efforts could occur between simulations due to fluctuations in cluster load and differences in configurations of the cluster nodes. Hence, each simulation is run for 3 times and the average CPU time is used in the studies. Output files are written in binary format of *vtk* for every process. Time taken to write the data files is also taken into consideration. The computational time is the maximum wallclock time for a single time step in Algorithm 4. The computational time on 10 processes is used as a baseline calculation. Speedup is measured as

$$\text{Speedup} = \frac{t_n}{t_p} \tag{5.1}$$

For the baseline calculation, $n = 10$ is used and t_p is the maximum wallclock time for a single time step with $p \geq n$.Efficiency is measured as

$$\text{Efficiency} = \frac{n \times t_n}{p \times t_p} \tag{5.2}$$

In this section, we will assess the strong scalability characteristics of our parallel approach. The studies are performed on the LUIS Cluster of Leibniz Universität Hannover using only Haswell-based nodes. Each Haswell-based node consists of two 8-core Intel Xeon E5-2630 processors. All nodes are interconnected with the Infiniband technology. Each sub-domain is assigned to one process (core).

5.2.1 Application to Taylor rod impact

The Taylor rod impact test is a widely accepted benchmark where a copper rod hits a rigid frictionless wall. The three-dimensional bar has a length of $L = 32.4$ mm and a circular cross-section with radius $R_0 = 3.2$ mm. The initial velocity is 227 m/s.

Material Model

In this benchmark problem, a finite plasticity material model with linear isotropic hardening is used to model the behavior of the rod. A detailed explanation can be found in (DE SOUZA NETO ET AL., 2008). The discretization through OTM framework in this section is similar to the discretization of equation of motion and the energy equation in Chapter
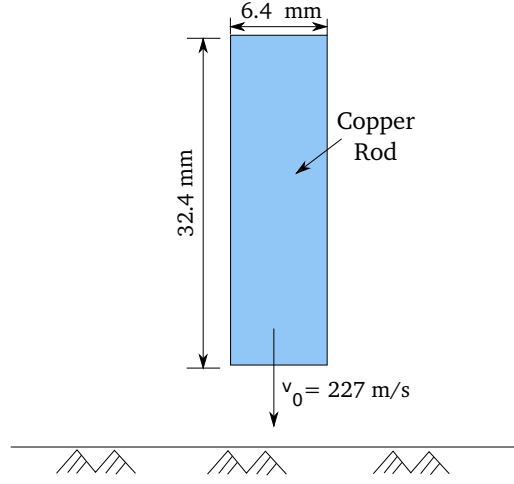
Figure 5.9. Geometrical setup of the Taylor rod test.

4. The formulation is based on the multiplicative split of the deformation gradient into an elastic and plastic part

$$\boldsymbol{F}_{pn} = \boldsymbol{F}^e_{pn}\boldsymbol{F}^p_{pn} \tag{5.3}$$

Assuming the strain behavior as Hencky strain measure, it can be expressed in terms of the left elastic Cauchy-Green strain tensor $\boldsymbol{b}^e_{pn}$ as

$$\boldsymbol{\varepsilon}^e_{pn} = \ln \mathbf{V}^e_{pn} = \ln \left(\sqrt{\mathbf{b}^e_{pn}} \right) \tag{5.4}$$

where $\mathbf{V}^e_{pn}$ is the elastic left stretch tensor.

For a given incremental deformation gradient $\Delta \boldsymbol{F}_{pn+1}$, the elastic trial left Cauchy Green tensor is computed as

$$\mathbf{b}^{e\,tr}_{pn+1} = \Delta \boldsymbol{F}_{pn+1}\mathbf{b}^e_{pn+1}\Delta \boldsymbol{F}^T_{pn+1} \tag{5.5}$$

Using the exponential map integrator, the Equation (5.4) can be expressed as

$$\boldsymbol{\varepsilon}^e_{pn+1} = \ln \left(\sqrt{\bar{\mathbf{b}}^{e\,tr}_{pn+1}} \right) - \frac{\gamma_{pn+1} - \gamma_{pn}}{\Delta t}\frac{\partial f_{pn+1}}{\partial \boldsymbol{\tau}_{pn+1}}, \quad \bar{\mathbf{b}}^{e\,tr}_{pn+1} = \mathbf{Q}_{pn+1}\mathbf{b}^{e\,tr}_{pn+1}\mathbf{Q}^T_{pn+1} \tag{5.6}$$

The elastic trial left Cauchy Green tensor, $\mathbf{b}^{e\,tr}_{pn+1}$, is transformed in the principal stress space using the rotation tensor $\mathbf{Q}_{pn+1}$.

The onset of plastic yielding is defined by the yield function f. The yield surface divides the elastic domain from the plastic domain and the Kirchhoff stresses $\boldsymbol{\tau}_{pn}$ must lie within the elastic domain or on the yield surface. von Mises plasticity model is used and its deviatoric part leads to plastic deformations

$$\boldsymbol{\tau}_{pn+1} = p\mathbf{I} + \boldsymbol{s}_{pn+1} = K\operatorname{tr}(\boldsymbol{\varepsilon}^e_{pn+1})\mathbf{1} + 2\mu \left(\boldsymbol{\varepsilon}^e_{pn+1} - \frac{1}{3}\boldsymbol{\varepsilon}^e_{pn+1} \cdot \mathbf{1} \otimes \mathbf{1} \right) \tag{5.7}$$

where, the constants K and μ are the compression modulus and the second Lamé constant.

To model large plastic deformations, the von -Mises yield criteria is applied alongwith linear isotropic hardening behavior (hardening modulus H)

$$f_{p\,n+1}^{tr} = \left\| 2\mu \boldsymbol{\varepsilon}_{p\,n+1}^{e\,tr} \right\| - 2\mu \Delta \gamma_{p\,n+1} - \sqrt{\frac{2}{3}} \left[\sigma_{Y_0} + H \left(\bar{\varepsilon}_{pn} + \sqrt{\frac{2}{3}} \Delta \gamma_{p\,n+1} \right) \right] \le 0 \quad (5.8)$$

where σ_{Y_0} corresponds to yield stress.

The evolution equation for the plastic strain in case of isotropic associated plasticity can be expressed in terms of Lie derivative of the elastic left Cauchy Green tensor

$$\mathcal{L}_v \boldsymbol{b}_{pn}^e = -2 \boldsymbol{d}_{pn}^p \boldsymbol{b}_{pn}^e = -2 \dot{\gamma}_{pn} \frac{\partial f_{pn}}{\partial \boldsymbol{\tau}_{pn}} \boldsymbol{b}_{pn}^e \quad (5.9)$$

where $\boldsymbol{d}_{pn}^p$ is the rate of plastic deformation, $\dot{\gamma}_{pn}$ is the rate of the plastic variable and f_{pn} is the yield function which is expressed in terms of norm of the deviatoric stress $\|\boldsymbol{s}_{pn}\|$ as $f_{pn} = \|\boldsymbol{s}_{pn}\| - \sqrt{\frac{2}{3}} \sigma_Y$. The plastic isotropy is modeled as $\boldsymbol{W}^p = 0$ with $\boldsymbol{W}^p$ as the skew symmetric part of the plastic velocity gradient.

For $f < 0$, the Kirchhoff stresses lie in the elastic domain. But, when $f > 0$ for $\Delta \gamma_{pn} = 0$, the yield criteria is violated and when needed, the plastic increment is adapted in order to fulfill the constraint $f = 0$ for Kirchhoff stresses to lie on the yield surface. This can be corrected using Equation (5.8) and the plastic increment can be computed as

$$\Delta \gamma_{p\,n+1} = \frac{f^{tr}}{2\mu + \frac{2}{3}H} \quad (5.10)$$

Subsequently, the Cauchy stress tensor can be updated as

$$\bar{\boldsymbol{\sigma}}_{p\,n+1} = J_{p\,n+1}^{-1} \left[K \operatorname{tr}(\boldsymbol{\varepsilon}_{p\,n+1}^{e\,tr})\mathbf{1} + 2\mu \left(\boldsymbol{\varepsilon}_{p\,n+1}^{e\,tr} - \frac{1}{3} \boldsymbol{\varepsilon}_{p\,n+1}^{e\,tr} \cdot \mathbf{1} \otimes \mathbf{1} \right) \right. \\ \left. - 2\mu \Delta \gamma_{p\,n+1} \frac{\boldsymbol{s}^{tr}}{\|\boldsymbol{s}^{tr}\|} \right] \quad (5.11)$$

where $\boldsymbol{s}^{tr} = 2\mu \left(\boldsymbol{\varepsilon}_{p\,n+1}^{e\,tr} - \frac{1}{3} \boldsymbol{\varepsilon}_{p\,n+1}^{e\,tr} \cdot \mathbf{1} \otimes \mathbf{1} \right)$ and through back transformation using the rotational tensors as in Equation (5.6), the Cauchy stress tensor can be written as

$$\boldsymbol{\sigma}_{p\,n+1} = \boldsymbol{Q}_{p\,n+1} \bar{\boldsymbol{\sigma}}_{p\,n+1} \boldsymbol{Q}_{p\,n+1}^T \quad (5.12)$$

The actual value of the Hencky strain can be obtained by correcting the trial values by using the plastic increment $\Delta \gamma_{p\,n+1}$ as

$$\boldsymbol{\varepsilon}_{p\,n+1}^e = \boldsymbol{\varepsilon}_{p\,n+1}^{e\,tr} - \Delta \gamma_{p\,n+1} \frac{\boldsymbol{s}^{tr}}{\|\boldsymbol{s}^{tr}\|} \quad (5.13)$$

Now, the actual elastic left Cauchy Green tensor can be computed in the principal strain space as

$$\bar{\mathbf{b}}_{p\,n+1}^e = \exp\left(2\boldsymbol{\varepsilon}_{p\,n+1}^e \right) \quad (5.14)$$

and it is transformed back into the deformed configuration as

$$\mathbf{b}_{p\,n+1}^e = \boldsymbol{Q}_{p\,n+1} \bar{\mathbf{b}}_{p\,n+1}^e \boldsymbol{Q}_{p\,n+1}^T \quad (5.15)$$

Contact Formulation

Additionally, a contact algorithm is needed to model the copper rod striking a rigid wall. A simple contact algorithm is used assuming that the wall is rigid and the tangential movement is frictionless. The normal gap $g_{I_{n+1}}$ of each node at the next time step can be computed as

$$g_{I\,n+1} = (\boldsymbol{x}_{I\,n+1} - \bar{\boldsymbol{x}}) \cdot \boldsymbol{n} \qquad (5.16)$$

where $\bar{\boldsymbol{x}}$ is the coordinate of the rigid plane and $\boldsymbol{n}$ is the normal vector on that rigid plane. To enforce the non-penetration condition, a Dirichlet boundary condition is applied on the corresponding node with prescribed displacements at the next time step

$$\boldsymbol{u}_{I\,n+1} = \boldsymbol{x}_{I\,n} - g_{I\,n+1}\boldsymbol{n} \qquad (5.17)$$

The above condition is only applied when the non-penetration condition is violated $g_{I\,n+1} < 0$. More details about formulations on two contacting deformable bodies can be found in (WRIGGERS, 2006).

Numerical Evaluation

The material parameters are chosen as $\nu = 0.35$ for the Poisson ratio, $E = 117.10^9 N/m^2$ for the Young's modulus, $\rho_0 = 8.93 \cdot 10^3 kg/m^3$ for the density, $H = 100.10^6 N/m^2$ for the hardening modulus and $Y_0 = 400.10^6 N/m^2$ for the initial yield stress. For a stable explicit time integration scheme, a computation step size of $\Delta t = 4.10^{-9}$ s is selected.

The initial domain is set up by triangulation with the material points located at the barycenters of the tetrahedral elements. Subsequently, the initial mesh is jettisoned and the computations proceed in a meshfree manner. The model contains 5,966 nodes and 28,423 material points. The domain decomposition is performed by distributing nodes and material points across all processes with the help of Zoltan library, see Section 5.1.1. Figure 5.10 shows a sequence of snapshots of the taylor rod impacting axially against a rigid boundary. Here, MPI Process Rank refers to the rank in order to identify a process, which is an integer in the range $[0, N-1]$ where N is total number of MPI processes.

Number of MPI processes	Number of nodes	Number of material points
50	120	569
100	60	285
150	40	190
199	30	143
239	25	119

Table 5.1. Taylor rod impact: Number of MPI processes and subdomain sizes for each process

The strong scaling studies are performed for upto 239 processes. It is evident from the Parallel Performance Analysis (Figure 5.11) that the speedup is almost super-linear till 150 processes and shows good strong-scalability for increasing number of processes. Since the geometry of the sub-domains are determined through RCB algorithm, the communication
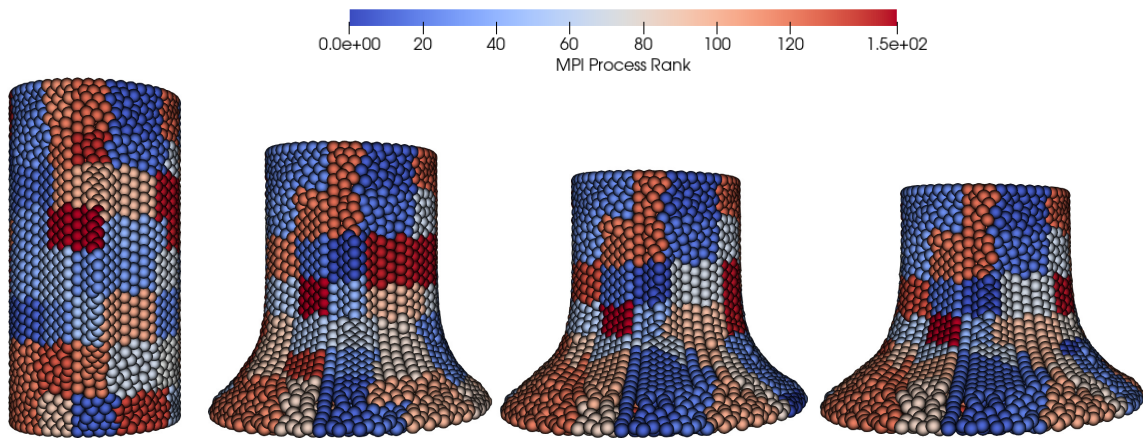
Figure 5.10. Snapshots of Taylor Rod deformation alongwith nodal distribution in different sub-domains

between the processes is minimized, leading to a high scalability. Also, with the use of suitable data management strategies as discussed in Section 5.1.3, overheads related to editing, insertion and deletion operations to the data structure has been minimized. With the use of nearest neighborhood communication patterns, it has been ensured that every sub-domain could have fixed maximum number of communications among its neighbors irrespective of the number of sub-domains.
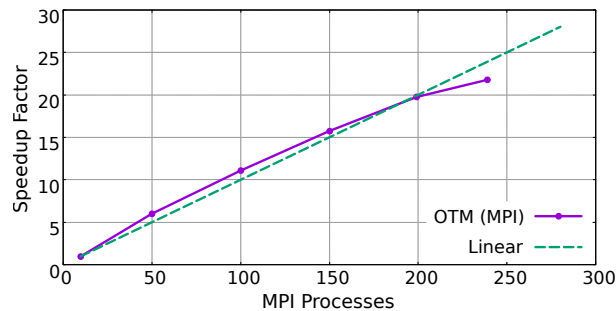

Figure 5.11. Parallel performance analysis: Strong scaling for Taylor rod impact.

Figure 5.12 shows a decrease in overhead communication and computation time for large process counts and smaller problems (sub-domains). With the increase in number of processes, the sub-domain size reduces (see Table 5.1). Thereby, the computation time for nodal and material point updates also decreases. Overall, it can be seen that high scalability has been obtained and the computational efforts of each subdomain and the communication costs decrease with an increase of number of processes. As the number of processes increase from 150 to 239, the communication time seems to stabilize, even though more data interfaces are present with the growth of subdomains (see Table 5.2). Once a sufficient number of subdomains are present (as seen for 100 to 239 processes), the communication costs of the exchanges between the neighbors are almost constant.
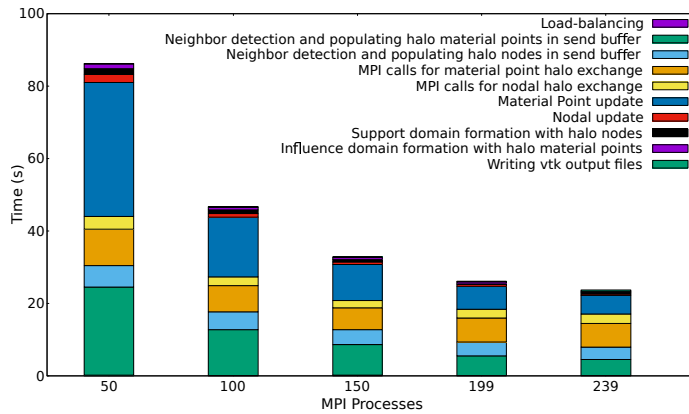
Figure 5.12. Taylor rod impact: Comparisons of growth in computational and com-
munication overhead time in strong scaling tests.

Number of MPI processes	Total halo material points	Total halo nodes
50	54,980	10,459
100	77,221	14,499
150	93,126	17,392
199	108,400	20,015
239	118,684	22,039

Table 5.2. Taylor rod impact: Number of MPI processes and total halo nodes and
material points at initial time step.

As shown in Table 5.3, the efficiencies are high and greater than 100% for some MPI pro-
cesses, which are similar to those as shown in the work by (LI ET AL., 2014). Also, super-
linear speedup has been achieved through our algorithm. This shows that with the better
implementation of the data structure, operations such as, editing, insertion and localised
search with linked cell method (Figure 5.8) has been handled effectively and the MPI over-
heads have been reduced leading to higher efficiencies. As shown in Figure 5.12, significant
amount of time is spent on neighbor detection and identifying halo nodes and material points,
which is a bottleneck in our implementation. Hence, a better implementation could be made
for the prediction of halo particles which are to be sent to the neighbors.

In order to compare the accuracy of the results obtained through the developed parallel OTM

Number of MPI processes	Wallclock time (s)	Speedup	Efficiency (%)
10	517.82	1	100
50	86.19	6.007	120.14
100	46.69	11.088	110.88
150	32.88	15.747	104.98
199	26.23	19.74	99.197
239	23.78	21.76	91.08

Table 5.3. Performance of the parallel implementation of the OTM framework for
the simulation of Taylor rod impact test.

	Final length (mm)	Final mushroom radius (mm)
(KAMOULAKOS, 1990)	21.47-21.66	7.02-7.12
(CAMACHO & ORTIZ, 1997)	21.42-21.44	7.21-7.24
(ERHART ET AL., 2006)	21.40-21.45	7.03-7.20
(LI ET AL., 2010)	21.43	6.8

Table 5.4. Taylor rod impact test: Comparison of results

framework with the ones available in the literature, the tailor rod impact test with the geometrical and material parameters as given in (WEISSENFELS & WRIGGERS, 2018) has been chosen as a benchmark. The mushroom of the rod at contact and the final length has been measured. The final mushroom radius to which the rod spreads at the impact surface is found to be 6.026 mm and the final length is 21.31 mm. The obtained final measurements are closer to the results which can be found in the literature (see Table 5.4).

5.2.2 Application to Serrated Chip Formation Process

In the second test case, numerical modeling of serrated chip formation is discussed. In this process, the material undergoes large plastic deformations. Beside physical mechanisms such as plastic deformations additionally adiabatic shear band formation and ductile fracture are involved. The separation of the chip from the workpiece is led by the ductile fracture in the cutting zone. Proper constitutive model and fracture model have to be applied to capture the physical phenomena involved in the chip formation process. Only basic equations are introduced and a more detailed explanation can be found in (HUANG ET AL., 2019).

Material Model

The formulation of the finite hyperelasto-plastic framework is based on local multiplicative decomposition of the deformation gradient into elastic and plastic part as shown in Equation (5.3). The plastic deformation is described by the Johnson-Cook flow stress model and the ductile fracture of the workpiece is described by the Johnson-Cook fracture model.
Assuming the dissipation potential $f = f(\boldsymbol{\tau}, R)$, the evolution equations have the following forms similar to Equation (5.9)

$$-\frac{1}{2}\mathcal{L}_v\boldsymbol{b}^e = \dot{\gamma}\frac{\partial f}{\partial\boldsymbol{\tau}}\boldsymbol{b}^e, \; \dot{\xi} = \dot{\gamma}\frac{\partial f}{\partial R} \tag{5.18}$$

where R is the thermodynamic force conjugate to the isotropic hardening variable ξ. The plastic multiplier γ has to fulfill the standard complementarity relation of Kuhn-Tucker conditions

$$\dot{\gamma} \geq 0, \; f(\boldsymbol{\tau}, R) \leq 0, \; \dot{\gamma}f(\boldsymbol{\tau}, R) = 0 \tag{5.19}$$

Using logarithmic strain measure (5.6) in the principal stress space, the linear stress strain relation can be applied to the elastic part of deformation

$$\boldsymbol{\tau} = \frac{\partial \psi}{\partial \varepsilon^e} = \lambda \operatorname{tr}(\varepsilon^e) \cdot \mathbf{1} + 2\mu\varepsilon^e, \tag{5.20}$$

where λ and μ are Lame constants and $\psi(\boldsymbol{b}^e, \varepsilon, \Theta)$ is the free energy. The evolution law (5.18) can be transformed into the evolution of logarithmic strain with the assumption of associated plastic law and by applying an exponential integrator,

$$\dot{\varepsilon}^p = \dot{\gamma} \frac{\partial f}{\partial \boldsymbol{\tau}}. \tag{5.21}$$

Using von Mises plasticity, the yield function is expressed in terms of Kirchhoff stress as

$$f^{p,\,flow}(\boldsymbol{\tau}) = \sqrt{\frac{3}{2}} \|\operatorname{dev}(\boldsymbol{\tau})\| - \sigma_Y(\varepsilon_{eq}^p, \dot{\varepsilon}_{eq}^p, \Theta) \tag{5.22}$$

where, $\boldsymbol{\tau}$ is the Kirchhoff stress, σ_Y the flow stress which is assumed to be a function of equivalent plastic strain rate $\dot{\varepsilon}_{eq}^p$, equivalent plastic strain ε_{eq}^p and the temperature Θ.
The equivalent plastic strain is defined as

$$\varepsilon_{eq}^p = \sqrt{\frac{2}{3}} \|\boldsymbol{\varepsilon}^p\|. \tag{5.23}$$

Subsequently, the equivalent plastic strain rate can be obtained as

$$\dot{\varepsilon}_{eq}^p = \dot{\gamma}. \tag{5.24}$$

The Cauchy stress $\boldsymbol{\sigma}$ can be obtained through back transformation as in Equation (5.12). Temperature increase occurs due to adiabatic heating from plastic deformation and the temperature evolution can be formulated as

$$\dot{\Theta} = \beta \frac{\sigma_v \dot{\gamma}}{\rho\, C_p}, \ \sigma_v = \sqrt{\frac{3}{2}} \|\operatorname{dev}(\boldsymbol{\sigma})\| \tag{5.25}$$

where, σ_v is von Mises equivalent stress, C_p is the heat capacity and β is the Taylor-Quinney coefficient.
Multiplicative decomposed power form of the flow stress has been applied to consider the effects of strain hardening, strain rate hardening and thermal softening. The Johnson-Cook hardening law (JOHNSON & COOK, 1983) is used to capture these effects

$$\sigma_Y = \left[A + B\left(\varepsilon_{eq}^p\right)^n\right]\left[1 + Cln\left(\frac{\dot{\varepsilon}_{eq}^p}{\dot{\varepsilon}_{e0}^p}\right)\right]\left[1 - \left(\frac{\Theta - \Theta_r}{\Theta_m - \Theta_r}\right)^m\right] \tag{5.26}$$

where A defines the initial yield stress, $\dot{\varepsilon}_{e0}^p$ is the reference plain strain rate, Θ_m is the melting temperature, Θ_r is the room temperature and, B, C, m and n are additional material parameters. Euler backward time integration scheme is used to solve the evolution equations based on the elastic predictor corrector return mapping algorithm, for more details, see (DE SOUZA NETO ET AL., 2008).

Separation of the chip from the workpiece and the serrated morphology on the chip upper surface can be described by the Johnson-Cook fracture model. At the vicinity of the tool

tip, high compression in the material and high concentration of strain occurs. Ductile fracture leads to separation of the material from the workpiece at the vicinity of the tooltip. Also, ductile fracture at the chip upper surface can lead to the formation of serrated chips. Johnson-Cook fracture model is used to model the ductile fracture and to predicts the fracture locations. When the accumulated equivalent plastic strain, $\varepsilon_{eq}^{p\,n}$ reaches the critical value,$\varepsilon_{eq\,f}^{p\,n}$, ductile fracture occurs

$$\varepsilon_{eq}^{p} \geq \varepsilon_{eq\,f}^{p} = [d_1 + d_2 \exp{(d_3\eta)}] \left[1 + d_4 ln\left(\frac{\dot{\varepsilon}_{eq}^{p}}{\dot{\varepsilon}_{e0}^{p}}\right)\right] \left[1 + d_5\frac{\Theta - \Theta_r}{\Theta_m - \Theta_r}\right] \tag{5.27}$$

where d_1, d_2, d_3, d_4 and d_5 are the material parameters, η is the stress triaxiality which is defined as

$$\eta = \frac{p}{\sigma_v}, \qquad p = \lambda\,\text{tr}\,(\varepsilon^e) \tag{5.28}$$

where p is the hydrostatic pressure and σ_v is the von Mises stress.

The deformations in the chip and the workpiece during metal cutting are driven by the cutting tool directly which moves in horizontal direction with a specific cutting depth and cutting speed. The non-penetration condition is defined by a projection of the slave node positions from the workpiece onto the cutting tool surface

$$g^N = (\boldsymbol{x}^s - \boldsymbol{x}^m) \cdot \boldsymbol{n}^m \geq 0 \tag{5.29}$$

The abbreviations g^N the normal gap, $\boldsymbol{x}^s$ the slave node from the workpiece $\boldsymbol{x}^m$ and $\boldsymbol{n}^m$ are the orthogonal projection of $\boldsymbol{x}^s$ on the tool surface and $\boldsymbol{n}^m$ is the normal vector associated to the tool body.

The normal contact force and the stick tangential contact force can be determined by using the penalty method as

$$\boldsymbol{t}^N = c_N\boldsymbol{g}_N, \qquad \boldsymbol{t}^T = c_T\boldsymbol{g}_T \tag{5.30}$$

where c_N and c_T are the penalty parameters.

The tangential contact force in the slip state is determined from the Coulomb friction law as

$$\boldsymbol{t}^T = -\mu\,\|\boldsymbol{t}_N\|\,\frac{\dot{\boldsymbol{g}}_T}{\|\dot{\boldsymbol{g}}_T\|} \tag{5.31}$$

where μ is the frictional coefficient. Further details can be found in (HUANG ET AL., 2019).

Numerical Evaluation

Ti6Al4V alloy is used as the workpiece material. The material parameters of the constitutive equations (5.26) and (5.27) can be found in (HUANG ET AL., 2019). The workpiece has a length and height of 300 μm and 120 μm respectively, see Figure 5.13. The cutting depth is 100 μm. The cutting tool is treated as rigid body with tool radius of 2 μm and rake angle of 0°. For the tool-chip contact modeling, the friction coefficient is set as 0.8. For the workpiece, the melting temperature Θ_m and the initial temperature Θ_r is set as 1630°C and 25°C respectively. For a stable explicit time integration scheme, a time step size of $\Delta t = 10^{-10}$ s is selected.
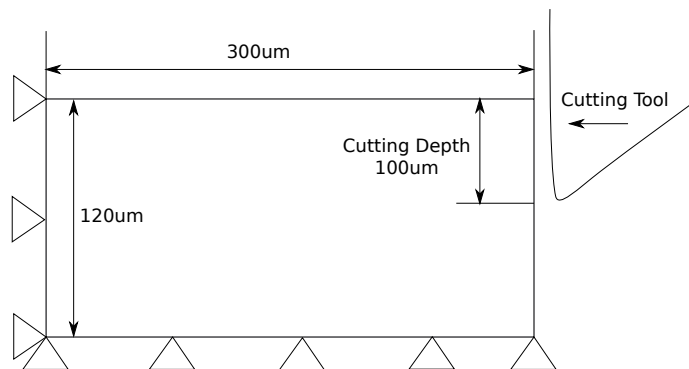
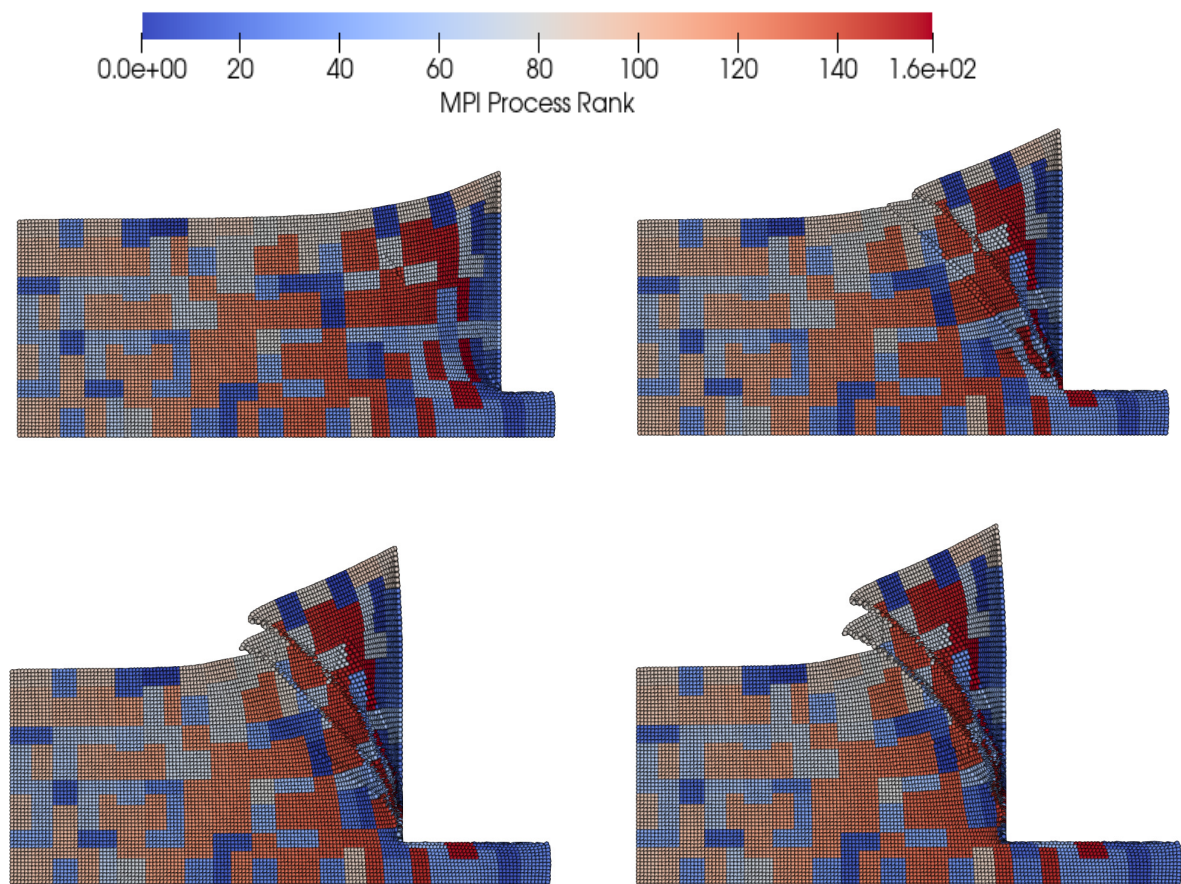**Figure 5.13.** Geometrical model for metal cutting.

Figure 5.14. Snapshots of serrated chip formation process alongwith nodal distribution in different sub-domains.

The model consists of 27,417 nodes and 107,975 material points. In this example, the scalability performance of the multiprocessing approach in the numerical solutions of large deformation problem is investigated. In Figure 5.14, the sequence of the serrated chip formation process is shown together with the corresponding nodal distribution across the sub-domains.
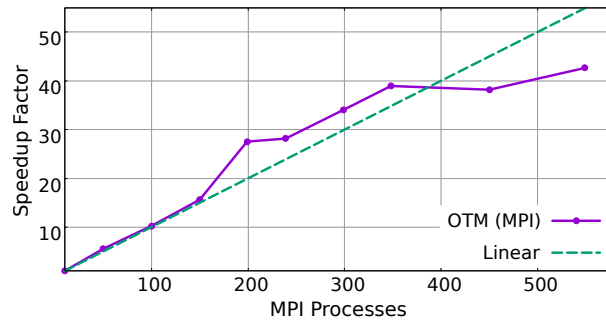
Figure 5.15. Parallel performance analysis: Strong scaling for serrated chip forma-
tion process.

Similar to the previous example, strong scaling studies are conducted for upto 549 processes. A substantial improvement in the speedup (Figure 5.15) can be observed leading to similar scalability as for the Taylor rod impact (Figure 5.11).

The communication overheads decrease with increase of processes, similar to that observed in Taylor rod impact example (Figure 5.16). In smaller problems, the ratio of particles (nodes and material points) in a sub-domain to halo particles is lower, which leads to an increase in halo communication overhead time compared to meaningful computation time. Due to the efficient implementation of nearest neighborhood communication and data structures, both the communication and computation costs decreases when additional computing resources are introduced. Similar to the Taylor rod impact example, with the reduction in sub-domain size (Table 5.5), reduction in computation cost is observed in this example (Figure 5.16). Additionally, the communication costs remain constant with increasing processes, even though the data interfaces along with halo particles increases (Table 5.6). This shows that our approach behaves in a optimized way and high efficiency has been achieved. These results make our approach a good competitor to (LI ET AL., 2014).
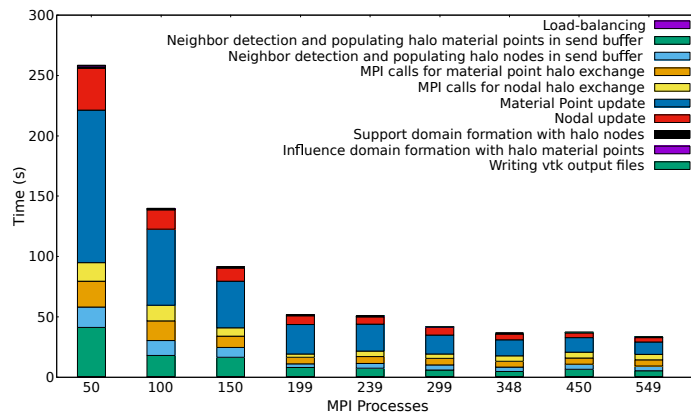

Figure 5.16. Serrated chip formation process: Comparisons of growth in computa-
tional and communication overhead time in strong scaling tests.

Number of MPI processes	Number of nodes	Number of material points
50	549	2160
100	275	1080
150	183	720
199	138	543
239	115	452
299	92	362
348	79	311
450	61	240
549	50	197

Table 5.5. Serrated chip formation process : Number of MPI processes and sub-domain sizes for each process.

Number of MPI processes	Total halo material points	Total halo nodes
50	20,855	4,903
100	29,304	7,628
150	37,450	9,403
199	44,680	11,275
239	48,743	12,134
299	55,204	14,217
348	59,121	15,083
450	67,821	17,400
549	75,151	19,667

Table 5.6. Serrated chip formation process: Number of MPI processes, total halo nodes, and material points at initial time step.

Number of MPI processes	Wallclock time (s)	Speedup	Efficiency (%)
10	1432.27	1	100
50	258.27	5.545	110.91
100	139.88	10.24	102.4
150	91.66	15.623	104.16
199	52.017	27.534	138.36
239	50.877	28.1518	117.79
299	42.063	34.0506	113.88
348	36.780	38.9414	111.90
450	37.520	38.1731	84.829
549	33.589	42.6405	77.669

Table 5.7. Performance of the parallel implementation of the OTM framework for the simulation of the serrated chip formation process.

Chapter 6

Parallel Multiphysics Simulation of Stereolithography Process

Stereolithography (SLA) is an additive manufacturing technique capable of producing highly accurate polymer components with high speed. This process is based on selective irradiation of photo-sensitive polymer resin by UV light in a layer-by-layer pattern. Capturing the relevant physics starts with a suitable UV irradiation model as discussed in Section 2.2. The evolution of the physics has been described schematically in four modular sub-models as shown in Figure 3.1. The input energy of UV irradiation induces a chemical photopolymerization reaction, leading to solidification of the fluid resin. The resin usually consists of one or more monomer(s)/oligomer(s) and photoinitiator, which controls the polymerization reaction.

The UV irradiation is a surface effect and only the nodes on upper layer of the resin are irradiated. The interaction of the UV light with the resin leads to the scattering phenomenon within the resin leading to inaccuracies in the polymerization process. However, due to constant UV intensity within the penetration depth, the scattering phenomenon is ignored (see modeling assumption in Section 3.1). In addition to assumptions mentioned in Section 3.1, for the purpose of simplicity, we consider only one Maxwell element for the simulation of the SLA process, although multiple Maxwell branches (see Figure 3.3) are required to accurately simulate the evolution of polymer network and resin properties. The elastic behavior is expressed in terms of the cure-dependent shear modulus as in Equation (3.67), which in turn makes the relaxation time cure-dependent. A summary of the implementation of equations for the SLA simulation can be found at the end of Chapter 3.

6.1 Implementation of the process simulation framework

In this work, the photopolymerization process for a single resin layer has been performed on a macro-scale length and the geometry domain being modeled has 5 mm thickness. The length and the width of the considered fluid resin is 20 mm and 20 mm respectively. The fluid resin is considered to be a limited section of the whole fluid resin whose vertical sides are allowed to deform freely. The displacements of the nodes are fully constrained (Dirichlet constraint, $\Delta = 0$) at the bottom of the resin at $z = 5$ mm to account for a proper bonding
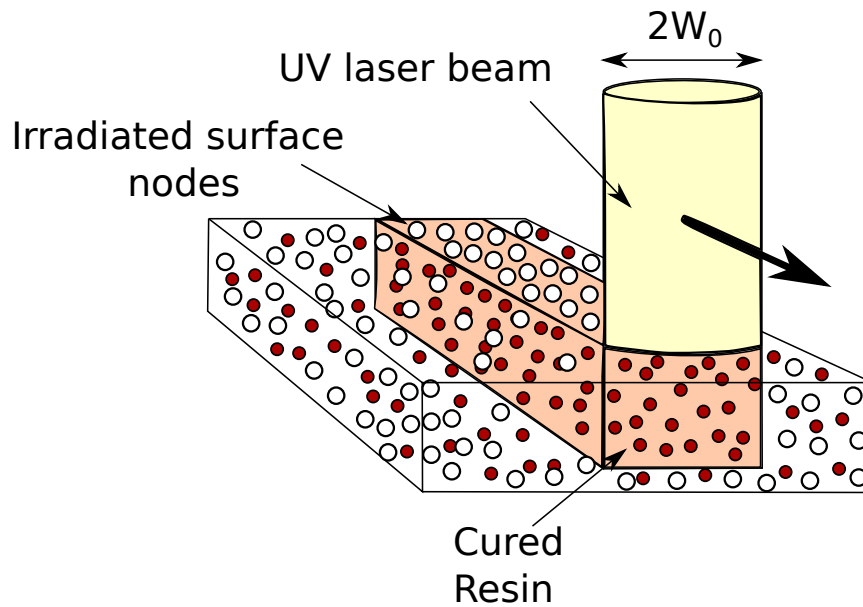
Figure 6.1. Schematic Representation of constant UV intensity along the resin depth, the surface nodes irradiated with UV laser and detection of the material points within the resin depth.

with the printing platform. Since the simulation is performed only on a limited section of the whole fluid resin, the lateral boundaries of the resin domain are allowed to deform freely. Also, the top and bottom surface of the domain will account for the convection boundary conditions. The initial resin temperature and the surrounding temperature is set at 300 K. The region of temperature variation is fairly small compared with the rest of the fluid resin. As heat conduction and/or molecular diffusion continues, this region will increase in size with time. The modeling has been performed within the OTM framework and the meshfree discretization of the equation of motion and the energy equation can be found in Chapter 4.

Material properties of DER 331 epoxy resin (by Dow Chemical Company) was used for the simulation. Some of the material parameters have been chosen from different sources due to the non-availability of these material parameters for DER 331 needed for our simulation. The fitting parameters are listed in Table 6.1. The curing related parameters for the cure reaction rate in Equation (2.5) are shown in Table 6.2. Although (KIM ET AL., 2015) has experimentally determined the curing related parameters for similar resins based on DER 331 epoxy resin, it would require approximately 10^{11} computation steps to cure at a particular laser position. Considering the constraints of limited computational resources, the parameters in Table 6.2 have been chosen such that the physical phenomena of high-speed curing can be captured within the simulation environment.

The laser follows a continuous circular path and during the process, the material points are exposed to laser multiple times (see Figure 6.2) which results in over-curing phenomenon. The total laser energy in Equation (2.2) at material points is the summation of the laser energies at every computation step due to continuous laser movement, see Figure 6.2. The driving force for the evolution of physics in the resin is the light intensity as shown in Figure 3.1. The detection of those material points which are affected due to laser irradiation within

Parameter	Value	Unit	Description	Source
ρ	1158.5	$[kg/m^3]$	Density	DER 331
η	12	$[Pa.s]$	Viscosity	DER 331
κ	0.17	$[W/m.K]$	Thermal Conductivity Coefficient	(DA SILVA BARTOLO, 2007)
h	0.002	$[W/m^2.K]$	Thermal Convection Coefficient	(DA SILVA BARTOLO, 2007)
$p_{\alpha_1}^{\mu}$	5.0	$[Pa]$	Shear modulus of uncured resin	(HARTMANN, 2019)
$p_{\alpha_2}^{\mu}$	613.7425	$[Pa]$	Shear modulus of cured resin	(HARTMANN, 2019)
ν	0.49	$[-]$	Poisson's Ratio	(HARTMANN, 2019)
β_c	-0.1	$[-]$	Chemical shrinkage parameter	(HARTMANN, 2019)
β_{Θ_f}	10^{-5}	$[1/K]$	Thermal expansion parameter	(HARTMANN, 2019)
β_{Θ_s}	10^{-5}	$[1/K]$	Thermal expansion parameter	(HARTMANN, 2019)

Table 6.1. Material properties used for simulations.

Parameter	A_{c1}	A_{c2}	B_1	B_2	m	n
Value	60000.0	18000.0	25.0	30.5	2.7	1.6
Unit	$[-]$	$[-]$	$[K]$	$[K]$	$[-]$	$[-]$

Table 6.2. Material parameters for the evolution equation of the degree of cure, cf. (HARTMANN, 2019)

the bulk of the resin is required, see Figure 6.1. In this work, due to laser movement, the spatial evolution of the curing process is represented while accounting for the photopolymerization kinetics. During photopolymerization, the variation of light intensity within the resin depth is provided as a modeling input, which means material points need to be detected according to varying light intensity within the resin. For the purpose of simplification, during the laser movement, the constant light intensity penetrates into the resin and forms a cylindrical region. Subsequently, a search process is used to detect the material points within this cylindrical region. These material points take part in the photopolymerization process. Simulation was performed with a laser power of 2W. Remaining parameters for laser modeling are shown in Table 6.3. For laser modeling, parameters such as D_p and E_c in Equation (2.3) are material specific properties. For the purpose of simulations, these experimentally-determined parameters were used for Gelatin methacrylate (GelMA) material from the work by (WADNAP ET AL., 2019).

The curing is restricted to laser exposed regions due to limited chemical diffusion/limited

Parameter	Value	Unit	Description	Source
P	2	$[W]$	UV Laser Power	-
R_{Laser}	0.005	$[m]$	Laser radius	-
W_0	0.002	$[m]$	Beam waist radius	-
D_p	0.7	$[-]$	Penetration depth coefficient	(WADNAP ET AL., 2019)
E_c	64,000	$[J/m^2]$	Critical energy needed for curing	(WADNAP ET AL., 2019)
T	0.00002	$[s]$	Laser Exposure Time	-

Table 6.3. Process parameters for UV laser used for simulations.

Figure 6.2. Laser path

dark cure potential as explained in Section 2.4. Hence, all particles exposed to UV laser have significantly higher degree of cure compared to the remaining particles in the fluid resin. The curing process provides a final conversion between 0 and the peak conversion.

Figure 6.3. Geometry of the fluid resin

Due to high-speed curing process, as discussed in Section 2.4, the curing process starts immediately as soon as the fluid resin is exposed to the UV light. The degree of cure evolves rapidly within fractions of a second of the UV laser exposure, which is a desirable phenomenon in the SLA process and the model captures this effect as shown in Figure 6.7(a). It can be seen the polymerization rate decreases rapidly at high conversion, which is at around 0.0001 s. As seen in Figure 6.1, since the UV intensity of the laser is uniform along the depth of the resin, curing is uniform along the entire depth of the resin. The evolution of monomer conversion is determined through the Arrhenius approach as in Equation (2.6). Unlike thermal curing, in UV curing process, the evolution of the degree of cure in the fluid resin is heavily dependent on the variation of laser intensity within the bulk of the fluid resin and the

evolution remains independent of the temperature increase (thermal diffusion) in the fluid resin. As a result, particles which are being irradiated by the laser have significantly higher degree of cure while rest of the resin remains in fluid state. A cross-sectional view of degree of cure evolution can be seen in Figure 6.5. The stages of evolution of the degree of cure (or photopolymerization process) during circular movement of laser is shown in Figure 6.4. To summarize, the UV laser-induced high-speed curing during SLA process is successfully simulated through the modeled moving UV laser.

(a)

(b)

(c)
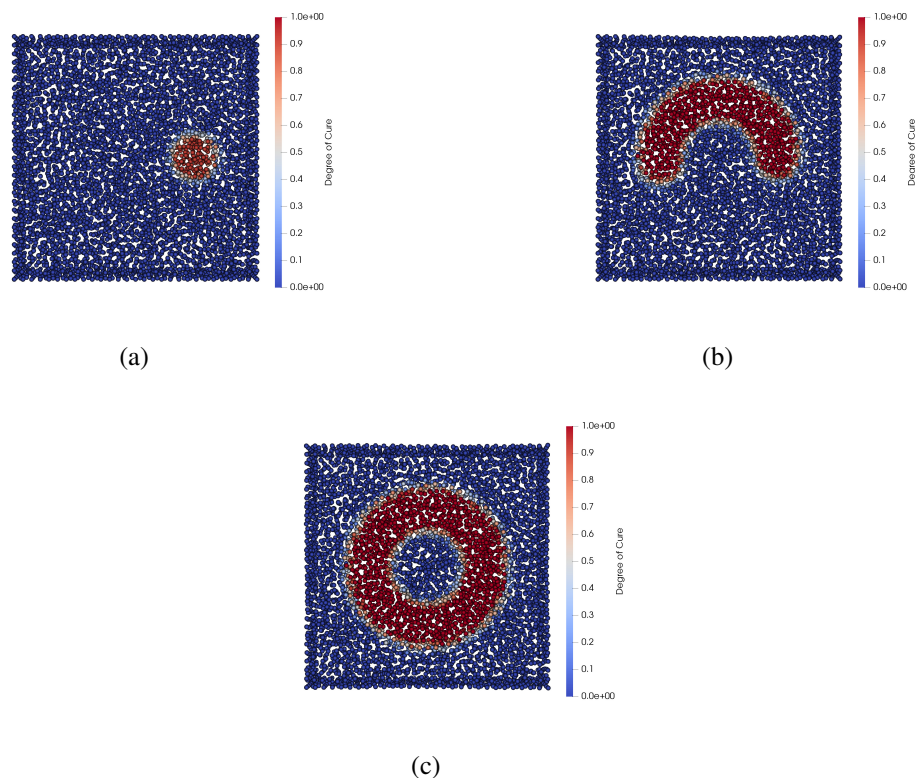
Figure 6.4. Evolution of Degree of Cure during laser movement on the top surface of the resin: (a) Curing process at the position of laser initiation. (b) Curing process when laser is mid-way. (c) Curing process when laser completes its circular movement and the final cylindrical cured part.

Evolution of the degree of cure is accompanied with an increase in temperature. Simulation was performed using a computation step of $\Delta t = 1.0 \cdot 10^{-7}$ and a final time of $t = 0.03s$. Based on the parameter ranges used in these simulations (see Table 6.1), the concentration of free radicals are significantly exhausted during curing and the temperature of the irradiated particles increases drastically within fraction of a second. A non-linear temperature increase is observed from 300K to 330K approximately. The rapid temperature rise is in line with the fact that the temperature rise is a result of the photopolymerization accompanied with an increase of degree of cure. The maximum temperature rise is observed when the resin is fully cured, see Figure 6.7(b). The computed temperature increase takes into account the process-dependant values of specific heat capacity and the thermo-chemical free energy function as
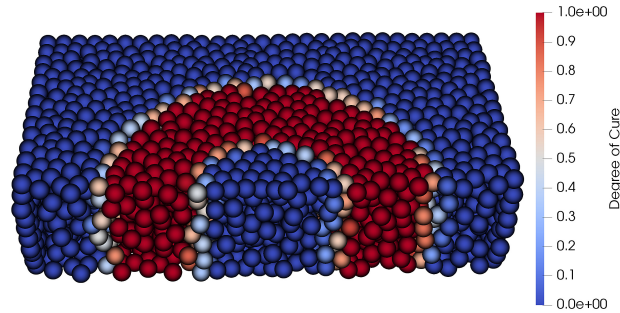
Figure 6.5. Cross-sectional view of the Degree of Cure evolution during the laser
movement

discussed in Section 3.3.1. Subsequently, after a particular location has been exposed to the laser, the temperature starts to decrease slowly because the reaction slows down and the heat conduction and convection starts to play an important role in dissipating the temperature rise. A convective heat transfer boundary condition at top and bottom surface of the resin is applied. Heat is leaving the cured region through heat conduction to the surrounding fluid region and through convection at the top and bottom surfaces of the resin. However, it takes some time for the heat to diffuse into rest of the fluid resin. Figure 6.6 shows the temperature variations at the location of laser initiation and it shows how the dissipation in temperature rise is progressing with time at that initial cured location during the complete laser movement. As a result, from Figure 6.6, larger dissipation can be observed in the top and bottom surface of the resin due to the role of convection. As the laser moves, temperatures at the initial point of laser irradiation starts to decrease and when the laser reaches back to the initial position, a small temperature rise is observed. After the laser movement ends and the laser is turned off, steady-state condition is reached and the temperatures at top, mid and bottom surfaces continue to decrease further, see Figure 6.6.

Figure 6.7(b) shows the exothermic nature of the photopolymerization process and a rapid temperature rise is accompanied with the evolution of degree of cure. By the time resin gets fully cured, maximum temperature is reached within the cured polymer. To illustrate the temperature evolution, the resulting temperature distribution considering the combined effects of laser irradiation, thermal diffusion and convection is depicted at different time instances in Figure 6.8. The temperature evolution is computed through the energy equation, whose discretized form has been presented in Equation (4.16). And, the temperatures at the nodes are updated accourding to Equation (4.30). During laser initiation, as in Figure 6.8(a), photopolymerization starts and temperature jump is observed due to exothermic process of heat release. As the UV laser progresses in its circular path, a similar circular like temperature profile is observed, see Figure 6.8(c). After completion of circular movement, laser is turned off and a steady state condition is observed, as seen in Figure 6.8(d) with further
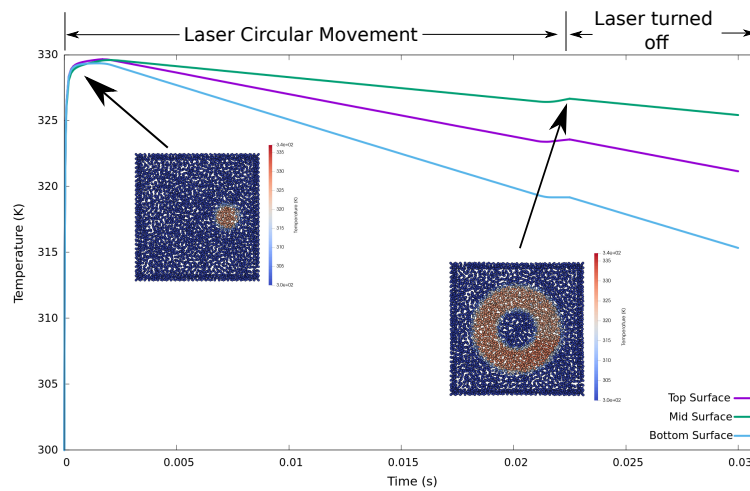
Figure 6.6. Temperature variation with time at the location of laser initiation on top, mid and bottom surface of the resin
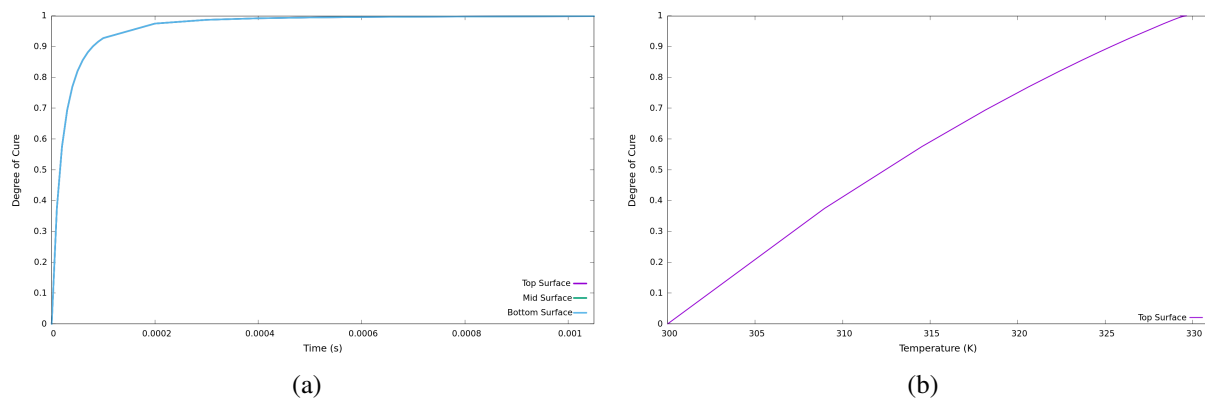

Figure 6.7. High Speed Curing: (a) Evolution of degree of cure within fractions of a second. (b) Temperature rise during photopolymerization process

temperature decrease on the already irradiated particles due to the role of combined effects of heat diffusion and convection, see Figure 6.6. Consequently, selected portions of the fluid resin has been subjected to UV laser irradiation at the considered time.

Volume shrinkage is a critical property in a polymerization process. Large volume shrinkages can cause distortion of the printed parts which in turn can affect the accuracy of the printed shape. The overall volumetric changes are a combined result of chemical shrinkage and thermal expansion. Chemical shrinkage is dependant on degree of cure and the thermal expansion has a coupled dependance on temperature and degree of cure. These volumetric changes are determined through functions as in Equation (3.66). Figure 6.11(a) shows the shrinkages occurring until the point where resin gets fully cured. The differences in shrinkages can be seen for top, mid and bottom surfaces of the printed part. Least shrinkage is observed for the bottom surface as a result of bonding between the printed part and the printing platform. In Figure 6.9(c), the shrinkages for the cured part during laser movement can be seen. These shrinkages depict the occurance of increase in density of the solidify-
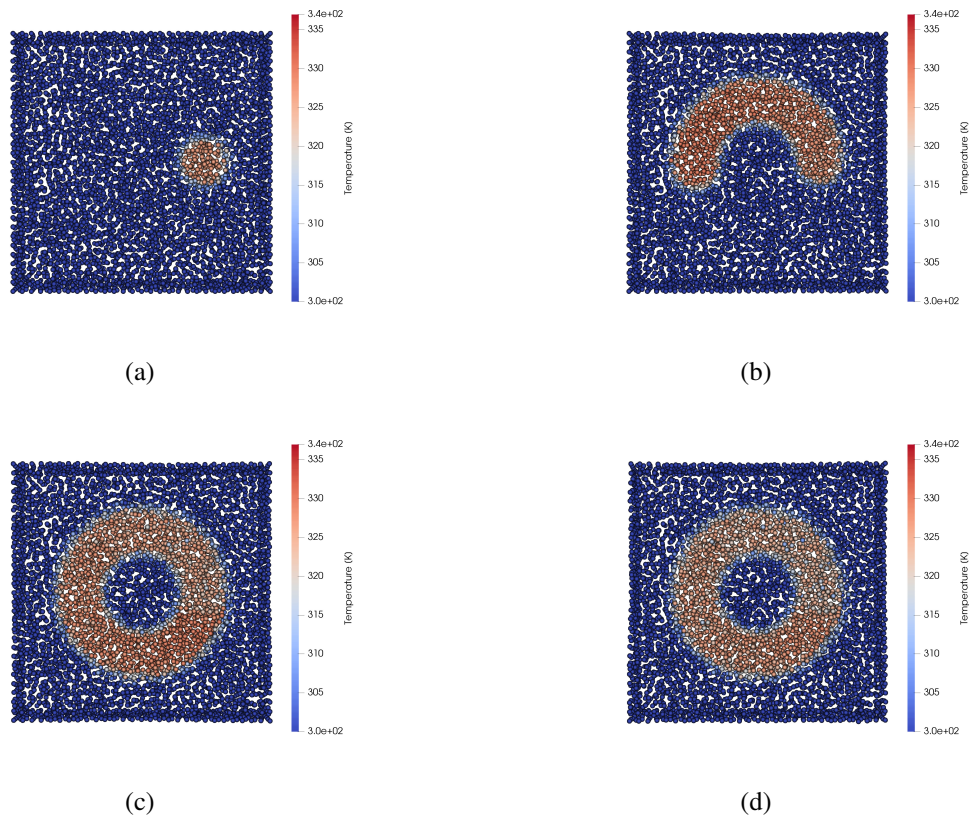
(a)

(b)

(c)

(d)

Figure 6.8. Temperature profile on the top surface of the resin at different time instances of photopolymerization: (a) Temperature distribution at laser initiation. (b) Temperature distribution when laser is mid-way. (c) Temperature distribution when laser completes its circular movement. (d) Temperature distribution when laser is turned off (steady-state condition)

ing resin due to increase in crosslinks of polymeric chains. A cross-sectional view if the shrinkages occuring during the printing process can be seen in Figure 6.10. Altogether, the effects of shrinkage and the subsequent temperature rise during photopolymerization process have been captured in the implemented framework replicating the physical processes involved. Also, it is quite interesting to see that the shrinkages and temperature changes are quite localized around the vicinity of the UV laser path.

Finally, in addition to volumetric changes, we also show the evolution of internal stress during the photopolymerization process. Usually, high internal stresses due to shrinkages in the printed part could be harmful for manufacturing of photo curing based applications, such as coatings. The evolution of internal stresses can be quantified into a single value through von Mises stress. The internal stress evolution for the top surface during the photopolymerization process is depicted in Figure 6.11(b). During photopolymerization, laser parameters, resin properties and curing kinetic parameters are responsible for the temperature increase. And this temperature increase influences the internal stress development associated with shrinkages in the cured part. These dependencies can be related by linking the Temperature model
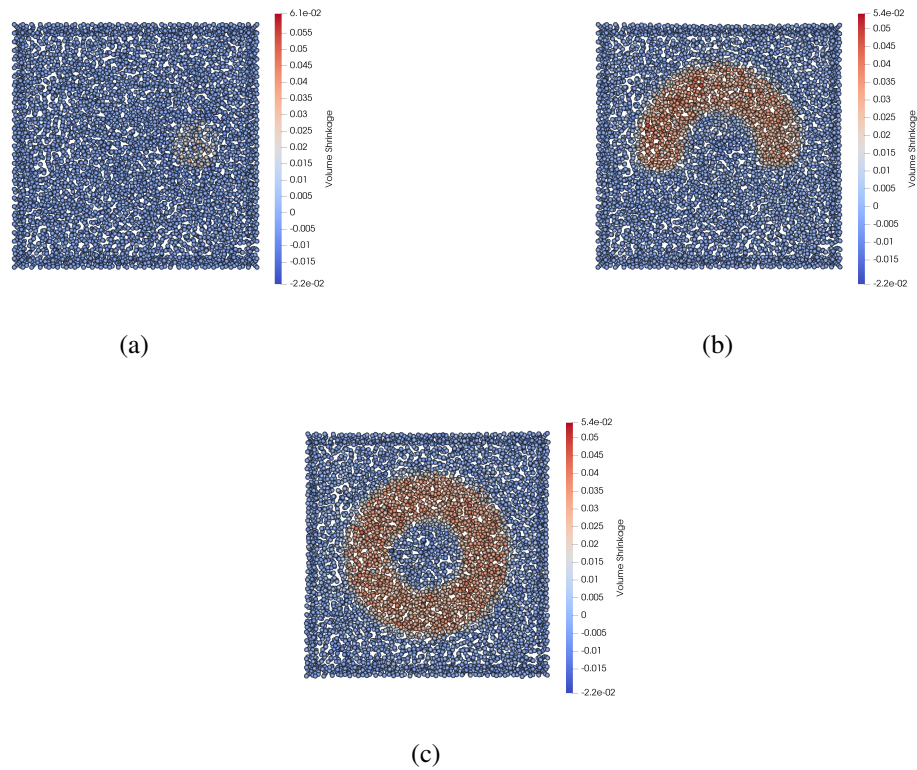
(a)

(b)

(c)

Figure 6.9. Volume shrinkage on the top surface of the resin during laser movement:
(a) at the position of laser initiation. (b) when laser is mid-way. (c) when
laser completes its circular movement.

and Material model in Figure 3.1. Maximum internal stresses are generated during the evolution of degree of cure and these stresses reach its peak when the material is fully cured. Then, slowly these internal stresses start to reduce. For instance, at a point on the resin surface which is initially exposed to UV Laser at the start of the printing process, the peak stress is 95.50Pa and by the time when the laser is turned off after completion of its circular movement, it reduces to 69.93Pa. Along the resin depth, the variations in final internal stresses for the top, mid and bottom surface is 69.93Pa, 38.38Pa and 33.64Pa respectively. Maximum internal stresses are generated at the top surface of the cured part and least stresses at the bottom of the cured part. In the work by (Wu ET AL., 2018), the simulations were conducted to study the evolution of mechanical properties in the photopolymer during photocuring process and similar trends of decreasing internal stress evolution along the resin depth have been observed. Variation of internal stresses during different time steps of laser movement and during the steady state can be seen from Figure 6.12. In Figure 6.12(a), high internal stresses can be seen at the point of laser initiation. Subsequently, as the laser moves to different position, as seen in Figure 6.12(b), the stresses at the point of laser initiation decreases with time. Finally, the end of the printing process, see Figure 6.12(d), stresses on the top surface can be seen to be almost uniformly distributed along the laser path.

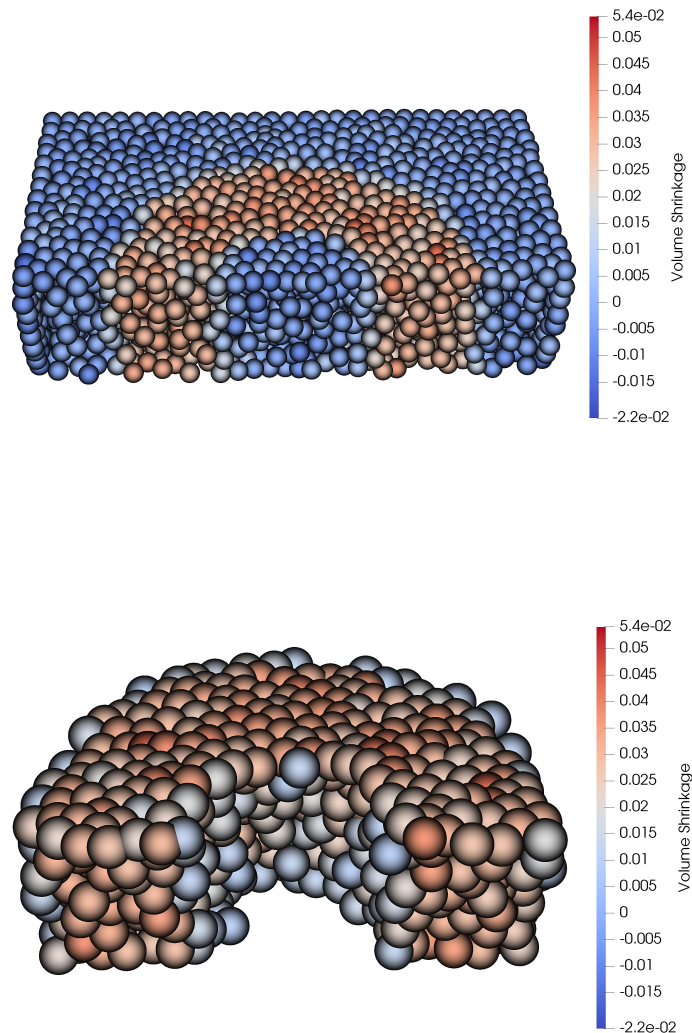The printed layer, ideally, should be a cylindrical region. But due to chemical shrinkage,

Figure 6.10. Cross-sectional view of the shrinkages within the printed part

thermal expansion and constraint on motion for the resin at the bottom of the plate, deformations occur. The fully cured layer after the circular movement of laser has been shown in Figure 6.13.

Finally, it is quite interesting to see how the mechanical, thermal and chemical properties evolve during the SLA process. During this process, a number of interesting effects accumulate. First, the degree of cure evolution induces chemical shrinkage in the resin following Equation (3.66). Additionally, the induced heat due to photopolymerization reaction results in thermal expansion according to Equation (3.66). The effect of conversion of degree of cure is the build-up of the stiffness, which is assumed to be linear as according to Equation (3.67). The phenomenon of high-speed UV curing process is captured within the performed simulation. The parameter characterization in this work is realized by identifying the model
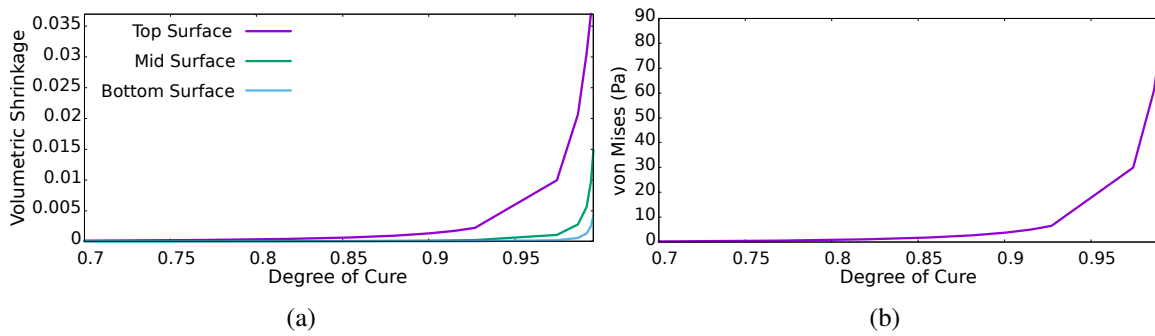
(a)

(b)

Figure 6.11. High Speed Curing: (a) Volume shrinkage with the evolution of degree
of cure. (b) internal stress evolution at the top surface caused by volume
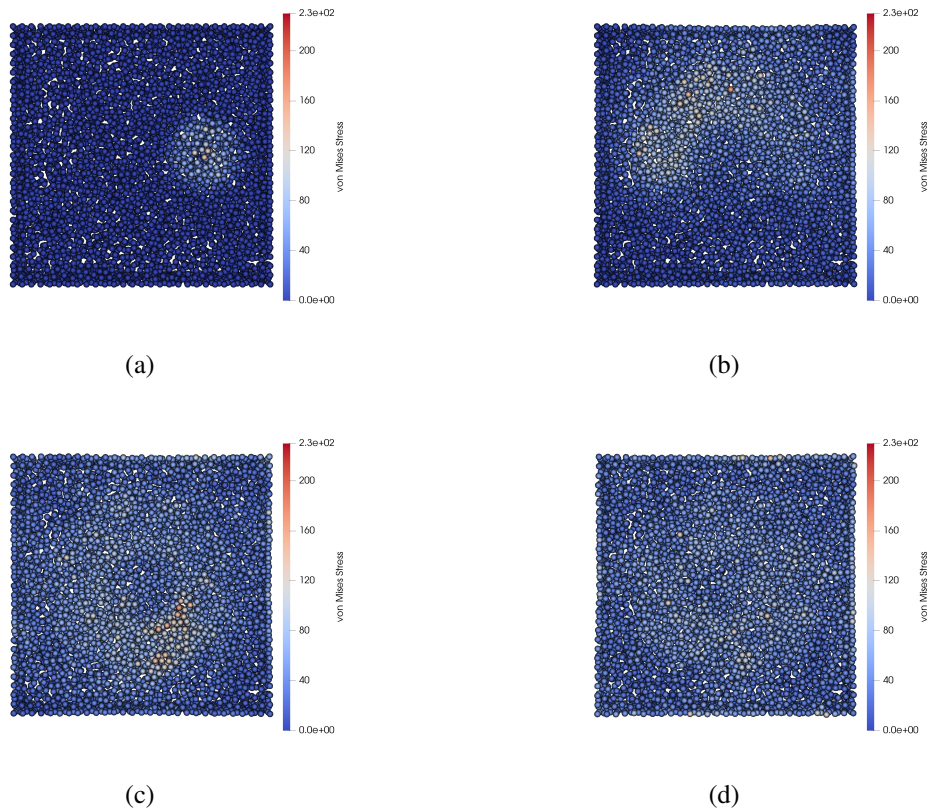shrinkage during photopolymerization.

(a)

(b)

(c)

(d)

Figure 6.12. Internal stresses on the top surface of the resin at different time instances
of photopolymerization: (a) Internal stress at the point of laser initia-
tion. (b) Internal stress distribution when laser is mid-way. (c) Internal
stress distribution when laser completes its circular movement. (d) Re-
duction in internal stresses when laser is turned off (steady-state condi-
tion)

parameters through numerical simulations. It means that the modeling parameters may not
be equal to the physical parameters and hence, validation of the model is qualitative in nature.

Hence, the aim is to identify whether the correct trends have been captured or not through the multi-physical approach. Consequently, the simulation model developed with the OTM framework is well-suited for the simulation of the SLA process and with the availability of sophisticated experimental data, it is worth to properly fit the developed material model. This will help to virtually optimize the SLA process within the developed model.
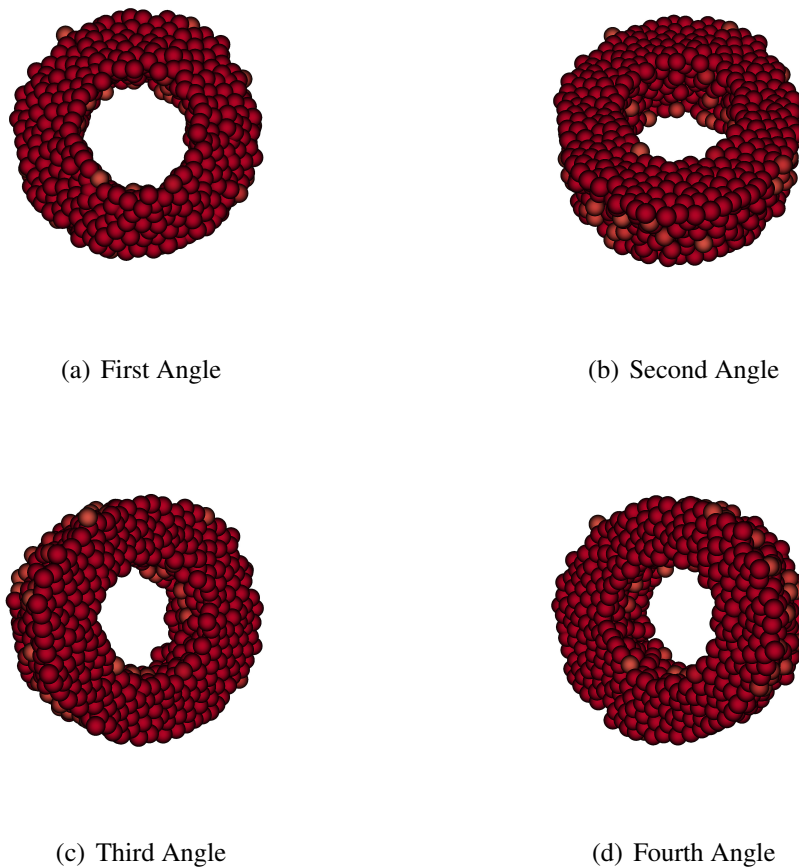

(a) First Angle

(b) Second Angle

(c) Third Angle

(d) Fourth Angle

Figure 6.13. Final shape of the printed layer

6.2 Parallel Performance

Parallel Performance of the SLA simulation process have been evaluated within the parallel framework and strong scalability characteristics have been studied. Same simulation conditions have been utilized to study the performance characteristics as mentioned in Section 5.2. The model of resin domain consists of 5,931 nodes and 26,574 material points. Strong scaling studies are performed on 49, 99 and 149 processes. As it can be seen in Figure 6.14, for large number of processes (149 processes), super-linear speedup has been achieved. For

the purpose of dynamic load-balancing, the Recursive Coordinate Bisection algorithm is called at optimized time intervals. Redistribution of nodes for the purpose of dynamic-load balancing along with the progress of photopolymerization can be seen in Figures 6.15 - 6.17.
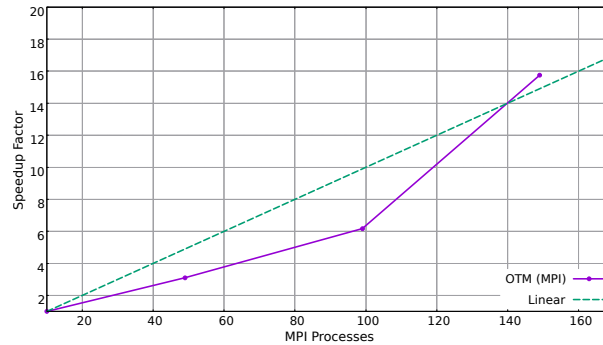

Figure 6.14. Parallel Performance Analysis: Strong Scaling for SLA Simulation Process

(a)   (b)
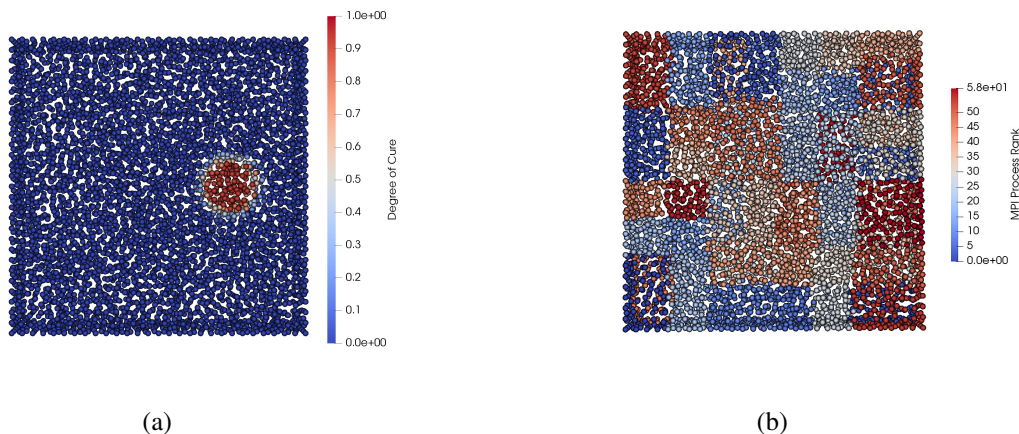
Figure 6.15. (a) Photopolymerization at the position of laser initiation. (b) Initial distribution of nodes among different MPI processes.

Figure 6.18 shows that the communication overheads decrease with increase of processes, similar to that observed in Taylor rod impact example (Figure 5.12) and in Serrated chip formation process example (Figure 5.16). Similarly, the effect of efficient implementation of nearest neighborhood communication and data structures can also be seen for SLA process. The communication cost seem to stabilize and remain constant when the number of process increase, see Figure 6.18, even though the data interfaces along with halo particles increases, see Table 6.5. Overall reduction in computational costs is observed with the reduction in sub-domain size (Table 6.4).

As it can be seen in Table 6.6, the efficiency of SLA simulations within the parallel framework is quite less compared to Taylor Rod (see Table 5.3) impact and Serrated chip formation process (see Table 5.7). This performance decrease could be due to the following reasons.
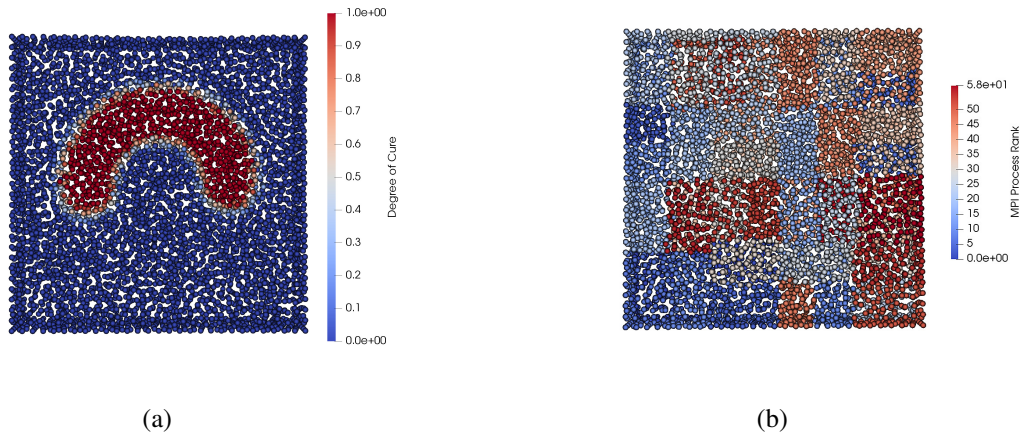
(a)

(b)

Figure 6.16. (a) Photopolymerization when laser is mid-way. (b) Redistribution of nodes among different MPI processes.
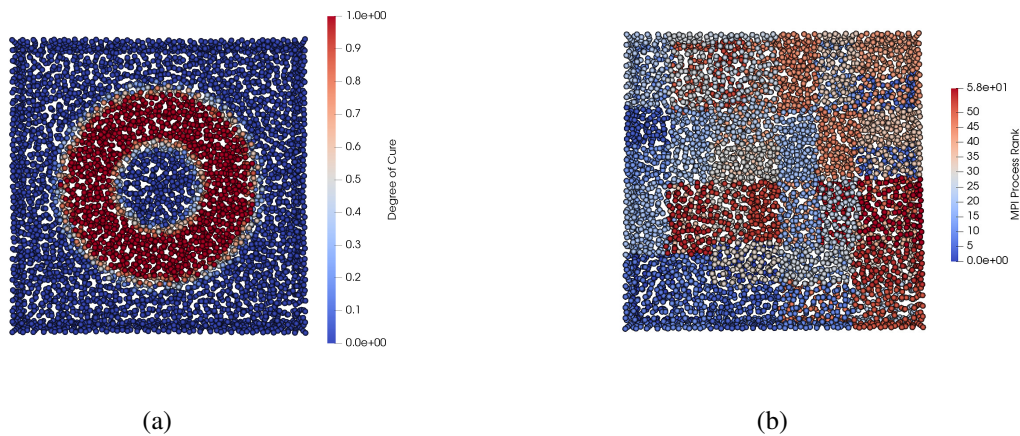

(a)

(b)

Figure 6.17. (a) Photopolymerization when laser completes its circular movement. (b) Redistribution of nodes among different MPI processes.

Number of MPI processes	Number of nodes	Number of material points
49	122	543
99	60	269
149	40	179

Table 6.4. SLA Simulationn Process : Number of MPI processes and Sub-domain sizes for each process

Most of the evolution takes place within the vicinity of the laser movement while rest of the fluid resin is barely evolving. Also, a search process is needed for material points influenced by laser irradiation and participating is the curing process. This search process takes place for all the material points in all the sub-domains instead of performing this search process
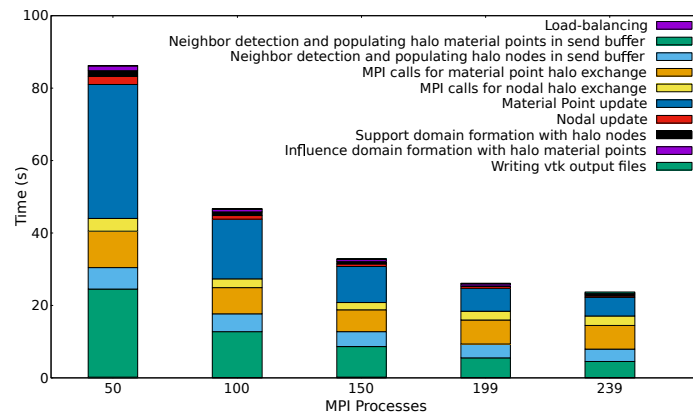
Figure 6.18. SLA Simulation Process: Comparisons of growth in computational and communication overhead time in strong scaling tests

Number of MPI processes	Total halo material points	Total halo nodes
49	43,023	8,741
99	62,879	12,701
149	78,002	15,463

Table 6.5. SLA Simulation Process: Number of MPI processes and total halo nodes and material points at initial computation step.

only in the vicinity of the laser beam. Hence, domain decomposition into higher number of sub-domains has almost no effect in reducing the computational load further. This shows that apart from reducing the sub-domain sizes, different computational processes which are being carried out have an effect on the parallel performance.

Number of MPI processes	Wallclock time (s)	Speedup	Efficiency (%)
10	283.356	1	100
49	91.20	3.1069	63.407
99	45.906	6.1725	62.348
149	40.1658	15.7475	47.346

Table 6.6. Performance of the parallel implementation of the OTM framework for the simulation of SLA Simulation Process

Chapter 7

Conclusions

A scalable OTM algorithm for large deformations, parallelized using MPI with an objective for scalability on large scale CPU clusters has been presented in this work. The consistency and robustness of this algorithm is demonstrated by two examples showing large deformation. Strong scaling studies were conducted. Implementation of dynamic halo regions have shown to improve the scalability by its ability to handle variable workloads and eliminating the storage issues related to fixed-size arrays. With the increase in number of processes, high scalability is observed. The communication costs decreases significantly and asymptotically even though more subdomain interfaces are present leading to increase in number of halo particles. The second advantage is the efficient data management strategy using advanced STL container adapted to fulfill various functionalities of data structure modifications. Flexible handling of data structures for two types of particles (nodes and material points) resulted in reduction of computational costs. Together with localized computation within each subdomain by using nearest neighborhood collectives for both nodes and material points this approach leads to high scalable results.

Stereolithography process is a complex process in the sense that several physical processes are involved therein. In this work, some of the key phenomena incorporated in the modeling framework are highly coupled thermo-chemo-mechanical evolution of resin properties and propagation of the UV laser through the resin. A classical continuum framework for the material modeling is introduced, which includes the related kinematics and the conservation principles. For the first time, to the best of author's knowledge, the modeling of the resin behavior including the classical models for the large strain viscoelasticity has been introduced. The light intensity initiates the photopolymerization and consequently generates heat due to its exothermic nature and also results in building up of mechanical stresses. In this work, a fully continuum based numerical scheme for the Stereolithography simulation was presented in the parallelized OTM framework. The method was motivated by the goal to exploit the meshfree features of the OTM method in order to handle several complex problems due to its multi-physical coupling during the photopolymerization process. The SLA process and the evolution of material properties has been modeled in a small time scale, which represents the actual physical phenomena of high-speed curing. In such a small time scale, the resin experiences changes in its thermal, chemical and mechanical properties, which has been captured in this model. Furthermore, the accuracy of the final printed part is determined through the

shrinkages and internal stresses generated during the printing process. The trends in variation of internal stresses was verified by the available data in the literature. The author believes that the developed simulation tool is novel and robust. To the best of author's knowledge, the novelity is due to the fact that the developed 3D simulation model gives the user full control to modify the material properties, shrinkages and stresses by providing full flexibility to tune up the modeling parameters. Also, the utilization of the OTM meshfree method is quite new in the field of Sterolithography simulation. Besides, its robustness originates from the fact the simulation tool is based on a parallellized meshfree framework, making it capable of handling highly complex multiphysical features of the printed part at a macro scale length. This can help in visualizing the actual printed parts. Nevertheless, the simulation tool enjoys the updated Lagrangian and meshless features of the OTM method.

In summary, this work presents a unified computational approach for the SLA process modeling based fully on the Optimal Transportation Meshfree (OTM) method. The distinctive characteristic of this research work is that it predicts the printed geometry on a macro scale length alongwith the inclusion of the deformations generated within it, achieved through multi-physical coupling. The field of Stereolithography simulation is still at the novice stage and the research in this area has lot of possibilities to further incorporate other aspects. Here are some directions for further researches as suggested by the author:

- To incorporate the experimental data into the developed simulation model

- To incorporate the Gaussian distribution of the laser energy within the fluid resin to capture the decrease in UV laser intensity with depth due to photo-absorption. Also, develop the related algorithm to identify particles exposed to this laser energy within the Gaussian distribution.

- To incorporate the multiple Maxwell branches in the rheological model in order to simulate the evolution of new crosslinks in addition to old crosslinked polymer network.

- To develop hybrid MPI-OpenMP parallel framework.

- To incorporate fault tolerance mechanisms in the parallel framework to help in overcoming instabilities while running large MPI jobs in clusters, which could be an asset in terms of savings in computational efforts.

Appendix A

Remarks on Parallel Framework

In the following chapter, additional parallel framework related implementations are introduced. To be more precise, communication buffers required for static and dynamic load-balancing and halo communication have been discussed. Also, the advantages of using specific container (STL map, see Section 5.7) in our data structure can be seen in the following chapter.

Static and dynamic load-balancing (Section 5.1.1) within the distributed memory framework is performed with the help of Zoltan library from Sandia Laboratories (BOMAN ET AL., 2007). This library, based on the geometrical locality of the particles (nodes and material points), provides a list of particles which needs to be migrated. Apart from this, zoltan library also performs the task of migrating the particles into their destination processes. This entire process of particle migration in the physical memory space ensures geometrical locality of the particles, i.e., particles which are physically close to each other in the spatial domain must be in the same process. But, in order to perform all these operations, the user must prepare the information needed for the migration: First of all, the particle information need to be removed from the sub-domain and put into the send buffer, which is then passed to the zoltan as input. Then, after zoltan performs the migration of particles, these particle information need to be merged (integrated) into the data structures of the sub-domain at the destination process.

The library takes the coordinate information of particles (nodes and material points) as input. The library is executed across all processes. Since particle information is read by all processes from input, each process stores its own set of particle information, see Figure A.1(a). Through internal communication of the library, particles are then assigned their corresponding destination processes, see Figure A.1(b). Particles are, then, moved to their destination processes by packing their information in a send buffer via user-defined data structures through Zoltan, see Figure A.2(a) and Figure A.2(b). At first, size of each particle (node or material point) are determined as shown in Listing A.1. Then, particle information is placed in the send buffer as shown in Listing A.2. After communication by Zoltan, the information is stored at receive buffer. Then, particle information from receive buffer is integrated into the data structure of the sub-domain at the destination process, see Figure A.2(b) and Listing A.3. All the above mentioned processes, such as, removing particle information from existing data structures at origin process, determining particle information size,

packing the information in send buffer and integrating the particle information into the data structures of destination process has to be performed by the user. Zoltan only assists in sending the information to the destination process. This entire process of particle migration in the physical memory space ensures geometrical locality of the particles, i.e., particles which are physically close to each other in the spatial domain must be in the same process, see Figure 5.1. It will also ensure the static and dynamic load-balancing among the sub-domains.
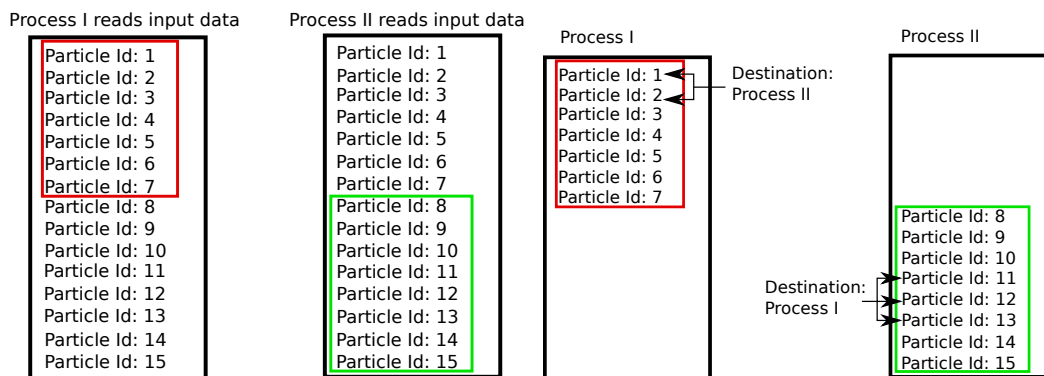
(a) Reading of Input File by all the processes and each process storing its own set of particle information

(b) Identification of destination process by Zoltan based on geometrical locality of particles

Figure A.1. Domain Decomposition by Zoltan [Steps: 1-2]
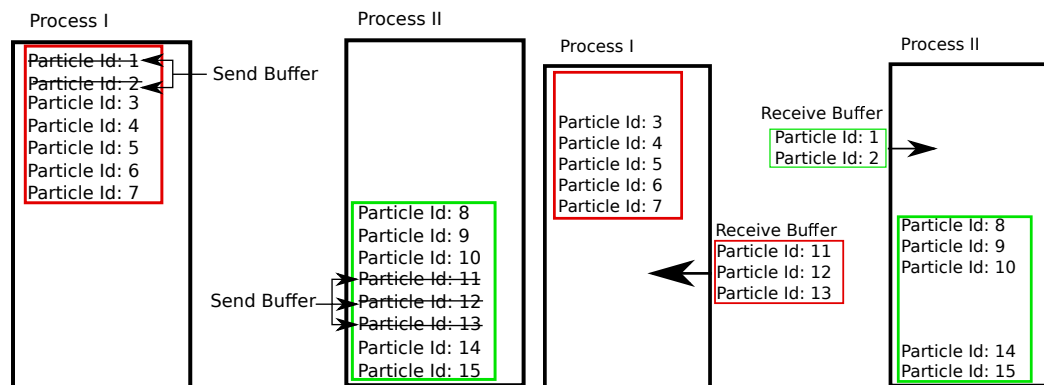
(a) Moving particle information into a send buffer

(b) Particle information being stored in receive buffer after internal communication by Zoltan

Figure A.2. Domain Decomposition by Zoltan [Steps: 3-4]

```
unsigned int Mpoint::give_buffer_size() const {
  return ((33 + 5 * support_and_shape.size()) * sizeof(double) +
    Particle::give_buffer_size());
}
```

Listing A.1. Code segment to identify buffer size of a material point.

```
unsigned int Mpoint::to_buffer(char *out) const {
  int pos = 0;
  pos += Particle::to_buffer(out);

  std::memcpy(out + pos, &volume, sizeof(double));
  pos += sizeof(double);   %Data is placed in buffer by increment of
      'double' to prevent any misalignment

  std::memcpy(out + pos, &rho, sizeof(double));
  pos += sizeof(double);
```

Listing A.2. Packing material point information in send buffer.

```
unsigned int Mpoint::from_buffer(char *in) {
    unsigned int pos = 0;

    pos += Particle::from_buffer(in);

    unsigned int tmp1, tmp2;

    std::memcpy(&volume, in + pos, sizeof(double));
    pos += sizeof(double);

    std::memcpy(&rho, in + pos, sizeof(double));
    pos += sizeof(double);
```

Listing A.3. Unpacking material point information from receive buffer.

```
 PARTICLE_TYPE *point_type = reinterpret_cast<PARTICLE_TYPE *>(&(
    recv_buffer_mp[done])); //Identify the particle type: Node or
    material point

if (*point_type == NEOHOOK) {
      unsigned int *gid = reinterpret_cast<unsigned int *>(&(
        recv_buffer_mp[done]) + sizeof(double)); // Extract the
        particle ID
      thelast.my_Mpoints[*gid] = std::make_shared<NeohookeanMpoint
        >(); //create a Particle object of NEOHOOK type

      thelast.my_Mpoints[*gid]->from_buffer(&recv_buffer_mp[done])
        ; //copy the particle information in the subdomain from
        the 'from_buffer'

...
...
```

Listing A.4. Integrating migrated particle information with the subdomain.

The nodes and material points are treated as particles, in general, by Zoltan with no distinction between them. While integrating the particle information into the data structures of the destination process from receive buffer, see Figure A.2(b), it is required to identify if the
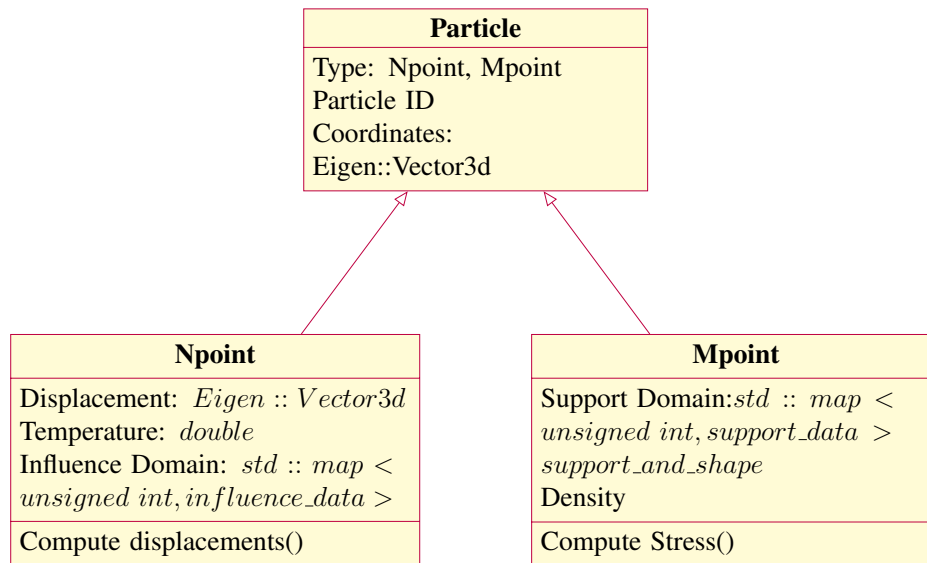
Figure A.3. Implementation of abstract classes

particle is a node or a material point with the help of Particle Type, see Listing A.4. Based on the Particle Type, which acts as a key, desired shared object (construction of an object owned by a shared pointer) for a node or material point (Figure A.3) is constructed at the destination process.

Bibliography

ARROYO M. & ORTIZ M. Local maximum-entropy approximation schemes: a seamless bridge between finite elements and meshfree methods. International Journal for Numerical Methods in Engineering, 65 (2006): 2167–2202.

BALAJI P., BUNTINAS D., GOODELL D., GROPP W. ET AL. Fine-Grained Multithreading Support for Hybrid Threaded MPI Programming. International Journal of High Performance Computing Applications (IJHPCA), 24(1) (2010): 49–57.

BARIGOU Y. & GABRIEL E. Maximizing Communication-Computation Overlap Through Automatic Parallelization and Run-time Tuning of Non-blocking Collective Operations. International Journal of Parallel Programming, 45 (2017): 1390–1416.

BARIGOU Y., VENKATESAN V. & GABRIEL E. Auto-tuning Non-blocking Collective Communication Operations. IEEE International Parallel and Distributed Processing Symposium Workshop, (2015): 1204–1213.

BARTOLO P. & MITCHELL G. Stereo-thermal-lithography: A new principle for rapid prototyping. Rapid Prototyping Journal, 9(3) (2003): 150–156.

BATHE K.J. Finite Element Procedures. Prentice Hall, 2006.

BELYTSCHKO T., KRONGAUZ Y., DOLBOW J. & GERLACH C. On the completeness of meshfree particle methods. International Journal for Numerical Methods in Engineering, 43 (1998) (5): 785–819.

BENNETT J. Measuring uv curing parameters of commercial photopolymers used in additive manufacturing. Additive Manufacturing, 18 (2017): 203–212.

BHATTACHARYA A. Radiation and industrial polymers. Progress in Polymer Science, 25(3) (2000): 371–401.

BOMAN E., DEVINE K., FISK L.A., HEAPHY R. ET AL. Zoltan 3.0: Parallel Partitioning, Load-balancing, and Data Management Services; Developer's Guide. Sandia National Laboratories, Albuquerque, NM (2007).

BOSSHARD C., BOUFFANAIS R., DEVILLE M., GRUBER R. ET AL. Computational performance of a parallelized three-dimensional high-order spectral element toolbox. Computers & Fluids, 44(1) (2011): 1–8.

CAMACHO G.T. & ORTIZ M. Adaptive Lagrangian modelling of ballistic penetration of metallic targets. Computer Methods in Applied Mechanics and Engineering, 142(3-4) (1997): 269–301.

CAO Z., PATRA A.K. & JONES M.D. Data Management and Volcano Plume Simulation with Parallel SPH Method and Dynamic Halo Domains. Procedia Computer Science, 108 (2017): 786–795.

CHAMBERS R., GUESS T. & HINNERICHS T. A phenomenological finite element model of part building in the stereolithography process. 1995.

CHEN M., ZHONG M. & JOHNSON J.A. Light-Controlled Radical Polymerization: Mechanisms, Methods, and Applications. Chemical Reviews, 116(17) (2016): 10 167–10 211.

CIARLET P.G. Mathematical Elasticity: Three Dimensional Elasticity. North-Holland, Amsterdam, New York, 1988.

CLASSENS K., HAFKAMP T., WESTBEEK S., REMMERS J.J.C. ET AL. Multiphysical modeling and optimal control of material properties for photopolymerization processes. Additive Manufacturing, 38 (2021): 101 520.

COLEMAN B.D. & NOLL W. The thermomechanics of elastic materials with heat conduction and viscosity. Archive for Rational Mechanics and Analysis, 13 (1963): 167–178.

CULLER D., KARP R., PATTERSON D., SAHAY A. ET AL. LogP: Towards a realistic model of parallel computation. In Proceedings of the Fourth ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming, pages 1–12. 1993.

DECKER C. High-speed curing by laser irradiation. Nuclear Instruments and Methods in Physics Research Section B: Beam Interactions with Materials and Atoms, 151 (1999): 22–28.

DECKER C. UV-radiation curing chemistry. Pigment & Resin Technology, 30(5) (2001): 278–286.

ERHART T., WALL W.A. & RAMM E. Robust adaptive remeshing strategy for large deformation, transient impact simulations. International Journal for Numerical Methods in Engineering, 65(13) (2006): 2139–2166.

FARHAT C. & ROUX F.X. A method of finite element tearing and interconnecting and its parallel solution algorithm. International Journal for Numerical Methods in Engineering, 32(6) (1991): 1205–1227.

FERRARI A., DUMBSER M., TORO E.F. & ARMANINI A. A new 3D parallel SPH scheme for free surface flows. Computers & Fluids, 38(6) (2009): 1203–1217.

FLACH L. & CHARTOFF R.P. A process model for nonisothermal photopolymerization with a laser light source. I: Basic model development. Polymer Engineering and Science, 35(6) (1995): 483–492.

FLANAGAN D.P. & BELYTSCHKO T. A uniform strain hexahedron and quadrilateral with orthogonal hourglass control. International Journal for Numerical Methods in Engineering, 17(5) (1981): 679–706.

FORUM M.P.I. MPI: A Message-Passing Interface Standard, Version 3.0. Message Passing Interface Forum, 2012.

FOUASSIER J.P. Photoinitiation, Photopolymerization, and Photocuring- Fundamentals and Applications. Hanser Publishers, 1995.

GHARBI Y.E., PARRET-FRÉAUD A., BOVET C. & GOSSELET P. Two-level substructuring and parallel mesh generation for domain decomposition methods. Finite Elements in Analysis and Design, 192 (2021): 103 484.

GHAZIMIRSAEED S.M., ZHOU Q., RUHELA A. & BAYATPOUR M. A hierarchical and load-aware design for large message neighborhood collectives. In Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis, pages 1–13. 2020.

GIBSON I., D.ROSEN & STUCKER B. Additive Manufacturing Technologies. Springer-Verlag New York, 2015.

GILLEN K. Effect of cross-links which occur during continuous chemical stress-relaxation. Macromolecules, 21(2) (1988): 442–446.

GRIEBEL M., KNAPEK S. & ZUMBUSCH G. Numerical Simulation in Molecular Dynamics: Numerics, Algorithms, Parallelization, Applications. Springer Science & Business Media, 2007.

HARTMANN P. Simulation of Thermo-Chemo-Mechanical Coupled Additive Manufacturing Processes using Peridynamics. Ph.D. thesis, Leibniz Universität Hannover, Germany (2019).

HOEFLER T., RABENSEIFNER R., RITZDORF H., DE SUPINSKI B.R. ET AL. The Scalable Process Topology Interface MPI 2.2. Concurrency and Computation: Practice & Experience, 23(4) (2011): 293–310.

HOEFLER T. & TRÄFF J.L. Sparse collective operations for MPI. In 2009 IEEE International Symposium on Parallel and Distributed Processing. IEEE, Rome, 2009.

HOLZAPFEL G.A. Nonlinear Solid Mechanics. A Continuum Approach for Engineering. John Wiley, Chichester, 2000.

HOLZAPFEL G.A. Nonlinear Solid Mechanics: A Continuum Approach for Engineering Science. Meccanica, 137 (2002): 489–490.

HOSSAIN M., POSSART G. & STEINMANN P. A small-strain model to simulate the curing of thermosets. Computational Mechanics, 43 (2009): 769–779.

HUANG D., WEISSENFELS C. & WRIGGERS P. Modelling of serrated chip formation processes using the stabilized optimal transportation meshfree method. International Journal of Mechanical Sciences, 155 (2019): 323–333.

HUANG Y., LEU M.C., MAZUMDER J. & DONMEZ A. Additive manufacturing: Current state, future potential, gaps and needs, and recommendations. Journal of Manufacturing Science and Engineering, 137(1) (2015): 014 001.

HUANG Y.M., KURIYAMA S. & JIANG C.P. Fundamental study and theoretical analysis in a constrained-surface stereolithography system. The International Journal of Advanced Manufacturing Technology, 24(5-6) (2004): 361–369.

HUGHES T. The Finite Element Method. Prentice Hall, Englewood Cliffs, New Jersey, 1987.

HUGHES T.J.R., LIU W.K. & ZIMMERMANN T.K. Lagrangian-Eulerian finite element formulation for incompressible viscous flows. Computer Methods in Applied Mechanics and Engineering, 29 (1981) (3): 329–349.

IHLEMANN J. Kontinuumsmechanische Nachbildung hochbelasteter technischer Gummiwerkstoffe. VDI-Verlag, Germany, 2003.

JACOBS P.F. & REID D.T. Rapid Prototyping & Manufacturing: Fundamentals of Stereolithography. Society of Manufacturing Engineers, 1992.

JARIWALA A.S., DING F., BODDAPATI A., BREEDVELD V. ET AL. Modeling effects of oxygen inhibition in mask-based stereolithography. Rapid Prototyping Journal, 17(3) (2011): 168–175.

JIANG C.P., HUANG Y.M. & LIU C.H. Dynamic finite element analysis of photopolymerization in stereolithography. Rapid Prototyping Journal, 12(3) (2006): 173–180.

JIN H., JESPERSEN D., MEHROTRA P., BISWAS R. ET AL. High performance computing using MPI and OpenMP on multi-core parallel systems. Parallel Comput., 37 (2011): 562–575.

JOHNSON G.R. & COOK W.H. A CONSTITUTIVE MODEL AND DATA FOR METALS SUBJECTED TO LARGE STRAINS, HIGH STRAIN RATES AND HIGH TEMPERATURES. In Seventh International Symposium on Ballistics. The Hague, The Netherlands, 1983.

KAMAL M.R. & SOUROUR S. Kinetics and thermal characterization of thermoset cure. Polymer Engineering and Science, 13(1) (1973): 59–64.

KAMOULAKOS A. A simple benchmark for impact. Bench Mark, (1990): 31–35.

KANG H.W., PARK J.H. & CHO D.W. Diffusion-limited photopolymerization in scanning micro-stereolithography. Applied Physics A: Materials Science and Processing, 79(8) (2004): 1839–1842.

KANG H.W., PARK J.H. & CHO D.W. A pixel based solidification model for projection based stereolithography technology. Sensors and Actuators A: Physical, 178 (2012): 223–229.

KIASAT M.S. Curing Shrinkage and Residual Stresses in Viscoelastic Thermosetting Resins and Composites. Ph.D. thesis, TU Delft, The Netherlands (2000).

KIM D., PARK H.J. & LEE K.Y. Study on curing behaviors of epoxy acrylates by UV with and without aromatic component. Macromolecular Research, 23(10) (2015): 944–951.

KIM D. & STANSBURY J. Extensive dark cure from controlled polymerization based on a method using visible-light activated initiator system. In AIChE Annual Meeting. 2008.

KIM K., SINHA J., GAO G., CHILDRESS K.K. ET AL. High-efficiency radical photopolymerization enhanced by autonomous dark cure. Macromolecules, 53(13) (2020): 5034–5046.

KOLMEDER S., LION A., LANDGRAF R., ET AL. Thermophysical properties and material modelling of acrylic bone cements used in vertebroplasty. Journal of Thermal Analysis and Calorimetry, 105(2) (2011): 705–718.

LANDGRAF R. Modellierung und Simulation der Aushärtung polymerer Werkstoffe. Ph.D. thesis, TU Cheminitz, Germany (2015).

LANDGRAF R., RUDOLPH M., SCHERZER R. & IHLEMANN J. Modelling and simulation of adhesive curing processes in bonded piezo metal composites. Computational Mechanics, 54 (2014): 547–565.

LAOIDE-KEMP C. Investigating MPI streams as an alternative to halo exchange. Ph.D. thesis, The University of Edinburgh, Edinburgh, U.K. (2015).

LAPIQUE F. & REDFORD K. Curing effects on viscosity and mechanical properties of a commercial epoxy resin adhesive. International Journal of Adhesion & Adhesives, 22(4) (2002): 337–346.

LASZLOFFY A., LONG J. & PATRA A.K. Simple data management, scheduling and solution strategies for managing the irregularities in parallel adaptive hp finite element simulations. Parallel Computing, 26(13-14) (2000): 1765–1788.

LEE J.H., PRUD'HOMME R.K. & AKSAY I.A. Cure depth in photopolymerization: Experiments and theory. Journal of Materials Research, 16(12) (2001): 3536–3544.

LI B., HABBAL F. & ORTIZ M. Optimal transportation meshfree approximation schemes for fluid and plastic flows. International Journal for Numerical Methods in Engineering, 83(12) (2010): 1541–1579.

LI B., STALZER M. & ORTIZ M. A massively parallel implementation of the Optimal Transportation Meshfree method for explicit solid dynamics. International Journal for Numerical Methods in Engineering, 100(1) (2014): 40–61.

LION A. & HÖFER P. On the phenomenological representation of curing phenomena in continuum mechanics. <u>Archives of Mechanics</u>, 59(1) (2007): 59–89.

LIU G.R. & LIU M.B. <u>Smoothed Particle Hydrodynamics: A Meshfree Particle Method</u>. World Scientific, 2003.

LONG K.N., DUNN M.L. & QI H.J. Mechanics of soft active materials with phase evolution. <u>International Journal of Plasticity</u>, 26(4) (2010): 603–616.

LUBARDA V.A. Constitutive theories based on the multiplicative decomposition of deformation gradient: Thermoelasticity, elastoplasticity, and biomechanics. <u>Applied Mechanics Reviews</u>, 57 (2004) (2): 95–108.

MAKNICKAS A., KAČENIAUSKAS A., KAČIANAUSKAS R., BALEVIČIUS R. ET AL. Parallel DEM Software for Simulation of Granular Media. <u>Informatica</u>, 17(2) (2006): 207–224.

MANDEL J. Balancing domain decomposition. <u>Communications in Numerical Methods in Engineering</u>, 9(3) (1993): 233–241.

MATIAS J.M., BARTOLO P.J. & PONTES A.V. Modeling and simulation of photofabrication processes using unsaturated polyester resins. <u>Journal of Applied Polymer Science</u>, 114(6) (2009): 3673–3685.

MELCHELS F.P.W., FEIJEN J. & GRIJPMA D.W. A review on stereolithography and its applications in biomedical engineering. <u>Biomaterials</u>, 31(24) (2010): 6121–6130.

MENDES-FELIPE C., OLIVEIRA J., ETXEBARRIA I., VILAS-VILELA J.L. ET AL. State-of-the-Art and Future Challenges of UV Curalbe Polymer-Based Smart Materials for Printing Technologies. <u>Advanced Materials Technologies</u>, 4(3) (2019).

MPI FORUM. <u>MPI: A Message-Passing Interface Standard, Version 3.1</u>. Message Passing Interface Forum, 2015.

NARAHARA H., TANAKA F., KISHINAMI T., IGARASHI S. ET AL. Reaction heat effects on initial linear shrinkage and deformation in stereolithography. <u>Rapid Prototyping Journal</u>, 5(3 (1999): 120–128.

NOTAY Y. & NAPOV A. A massively parallel solver for discrete poisson-like problems. <u>Journal of Computational Physics</u>, 281 (2015): 237–250.

OGER G., LE TOUZÉ D., GUIBERT D., DE LEFFE M. ET AL. On distributed memory MPI-based parallelization of SPH codes in massive HPC context. <u>Computer Physics Communications</u>, 200 (2016): 1–14.

PATRA A.K., BAUER A.C., NICHITA C.C., PITMAN E.B. ET AL. Parallel adaptive numerical simulation of dry avalanches over natural terrain. <u>Journal of Volcanology and Geothermal Research</u>, 139(1-2) (2005): 1–21.

PERRY M.F. & YOUNG G.W. A mathematical model for photopolymerization from a stationary laser light source. Macromolecular Theory and Simulations, 14(1) (2005): 26–39.

PHAM D.T. & GAULT R.S. A comparison of rapid prototyping technologies. International Journal of Machine Tools and Manufacture, 38(10-11) (1998): 1257–1287.

PLIMPTON S.J. & DEVINE K.D. MapReduce in MPI for Large-scale graph algorithms. Parallel Computing, 37(9) (2011): 610–632.

PRIME R.B. Thermosets. In: E. A. Turi (ed) Thermal Characterization of Polymeric Materials. Academic Press, London, 1997.

PRIMS O.T., CASTRILLO M., ACOSTA M.C., MULA-VALLS O. ET AL. Finding, analysing and solving MPI communication bottlenecks in Earth System models. Journal of Computational Science, 36 (2019).

SCHNABEL W. Polymers and Light: Fundamentals and Technical Applications. Wiley-VCH Verlag GmbH & Co. KGaA, 2007.

SELLI E. & BELLOBONO I. Photopolymerization of multifunctional monomers: kinetic aspects. In: J.-P. Fouassier and J. F. Rabek (eds) Radiation Curing in Polymer Science and Technology. Springer Netherlands, 1993.

SELVAM M. & HOFFMANN K.A. MPI/Open-MP hybridization of higher order WENO scheme for the incompressible Navier-Stokes equations. In AIAA SciTech. 2015.

SHANG Z. High performance computing for flood simulation using Telemac based on hybrid MPI/OpenMP parallel programming. International Journal of Modeling, Simulation, and Scientific Computing, 5(4) (2011).

DA SILVA BARTOLO P. Photo-curing modelling: Direct irradiation. The International Journal of Advanced Manufacturing Technology, 32 (2007): 480–491.

DA SILVA BARTOLO P.J. Optical Approaches to Macroscopic and Microscopic Engineering. Ph.D. thesis, University of Reading, United Kingdom (2001).

SOUROUR S. & KAMAL M.R. Differential scanning calorimetry of epoxy cure: isothermal cure kinetics. Thermochimica Acta, 14(1-2) (1976): 41–59.

DE SOUZA NETO E.A., PERIĆ D. & OWEN D.R.J. Computational Methods for Plasticity : Theory and Applications. John Wiley & Sons, Chicester, 2008.

SULSKY D., CHEN Z. & SCHREYER H.L. A particle method for history- dependent materials. Computer Methods in Applied Mechanics and Engineering, 118(1-2) (1994): 179–196.

TANG Y. Stereolithography Cure Process Modeling. Ph.D. thesis, Georgia Institute of Technology, Atlanta, GA (2005).

TARABEUX J., PATELOUP V., MICHAUD P. & CHARTIER T. Development of a numerical simulation model for predicting the curing of ceramic systems in the stereolithography process. Journal of the European Ceramic Society, 38(11) (2018): 4089–4098.

WADNAP S., KRISHNAMOORTHY S., ZHANG Z. & XU C. Biofabrication of 3D cell-encapsulated tubular constructs using dynamic optical projection stereolithography. Journal of Materials Science: Materials in Medicine, 30(3) (2019).

WANG B., HU Z. & ZHA G.C. General Subdomain Boundary Mapping Procedure for Structured Grid Implicit CFD Parallel Computation. Journal of Aerospace Computing, Information and Communication, 5(11) (2008): 425–447.

WANG X., JIANG M., ZHOU Z., GOU J. ET AL. 3D printing of polymer matrix composites: A review and prospective. Composites Part B: Engineering, 110 (2017): 442–458.

WEISSENFELS C. Simulation of Additive Manufacturing Using Meshfree Methods: With Focus on Requirements for an Accurate Solution. Springer-Verlag, 2021.

WEISSENFELS C. & WRIGGERS P. Stabilization Algorithm for the Optimal Transportation Meshfree Approximation Scheme. Computer Methods in Applied Mechanics and Engineering, 329 (2018): 421–443.

WESSELS H., BODE T., WEISSENFELS C., WRIGGERS P. ET AL. Investigation of heat source modeling for selective laser melting. Computational Mechanics, 63 (2019): 949–970.

WESSELS H., WEISSENFELS C. & WRIGGERS P. Metal particle fusion analysis for additive manufacturing using the stabilized optimal transportation meshfree method. Computer Methods in Applied Mechanics and Engineering, 339 (2018): 91–114.

WESTBEEK S., VAN DOMMELEN J.A.W., REMMERS J.J.C. & GEERS M.G.D. Multiphysical modeling of the photopolymerization process for additive manufacturing of ceramics. European Journal of Mechanics: A/Solids, 71 (2018): 210–223.

WESTBEEK S., REMMERS J.J.C., VAN DOMMELEN J.A.W. & GEERS M.G.D. Multi-scale process simulation for additive manufacturing through particle filled vat photopolymerization. Computational Materials Science, 180 (2020).

WRIGGERS P. Computational Contact Mechanics. Springer-Verlag, Berlin, Heidelberg, 2006, 2nd edition.

WRIGGERS P. & AVCI B. Discrete element methods: Basics and applications in engineering. In DE LORENZIS L. & DÜSTER A., eds., Modeling in Engineering Using Innovative Numerical Methods for Solids and Fluids, volume 599, pages 1–30. CISM, CISM International Centre for Mechanical Sciences, Courses and Lectures, 2020.

WU J., ZHAO Z., HAMEL C.M., MU X. ET AL. Evolution of material properties during free radical photopolymerization. Journal of the Mechanics and Physics of Solids, 112 (2018): 25–49.

YANG E., BUI H.H., STERCK H.D., NGUYEN G.D. ET AL. A scalable parallel computing SPH framework for predictions of geophysical granular flows. Computers and Geotechnics, 121 (2020).

YEBI A. & AYALEW B. Optimal layering time control for stepped- concurrent radiative curing process. Journal of Manufacturing Science and Engineering, 137(1) (2015).

ZETTERLUND P.B. & JOHNSON A.F. Free volume-based modelling of free radical crosslinking polymerisation of unsaturated polyesters. Polymer, 43(7) (2002): 2039–2048.

ZHAO L. & HU X. Autocatalytic curing kinetics of thermosetting polymers: A new model based on temperature dependent reaction orders. Polymer, 51(16) (2010): 3814–3820.

ZHOU C., YE H. & ZHANG F. A novel low-cost stereolithography process based on vector scanning and mask projection for high-accuracy, high-speed, high-throughput, and large-area fabrication. Journal of Computing and Information Science in Engineering, 15(1) (2015).

Curriculum vitae

Sandeep Kumar
born March 31, 1991 in Rourkela, India

Professional Experience

since 05.2016	Research assistant at the Institute of Continuum Mechanics, Gottfried Wilhelm Leibniz Universität Hannover, Germany
10.2017 - 12.2017 09.2018 - 11.2018 04.2019 - 05.2019	Visting researcher at the Laboratoire de Méchanique et Technology, Ecole Normale Supérieure Paris-Saclay, France
09.2019 - 12.2019 06.2019 - 07.2019	Visting researcher at the Laboratoire de Mécanique Multiphysique Multiéchelle (LaMcube), Université de Lille, France

Education

08.2013 - 06.2015	Indian Institute of Technology (IIT) Guwahati, India Degree: Master of Technology (Mechanical Engineering) Focus: Computational Mechanics
07.2009 - 06.2013	Kalinga Institute of Industrial Technology (KIIT), India Degree: Bachelor of Technology (Mechanical Engineering)

Scholarships

2021	Promotionsabschlussstipendium from Graduiertenakademie, Leibniz Universität Hannover
2019	Scholarship from French-German Doctoral College Sophisticated Numerical and Testing Approaches (SNTA)
08.2013 - 06.2015	Scholarship for Master's program, Indian Institute of Technology (IIT) Guwahati, India