Survey on user centric approaches to various pre-fracturing techniques in different materials in Houdini

Leah Elizabeth Sreshta

MSc Computer Animation and Visual Effects

S5104397

ACKNOWLEDGMENTS

I would like to express my deep sense of gratitude to Mr. Jon Macey, Bournemouth University and Mr Richard Southern, Head of the NCCA for encouraging me to pursue my final thesis at Bournemouth University.

I would also like to thank Prof. Jian Chang, Prof. Jian Zhang and Prof. Lihua You, faculty of Bournemouth university for supporting me over these two months and giving me valuable insights for this project. I would additionally like to thank Mr. Phil Spicer for all his help and valuable knowledge in Houdini and giving me a great learning experience.

I am also very grateful to my course mates Alec Watkins, Ben Carey, Bianca Farquharson, Clarence Rajaratnam, Daria Koslova, James Preston, Jamie Slowgrove, Rachael Strohkorb and Thomas Ashby for their valuable inputs, suggestions and constructive criticism which contributed immensely to the evolution of ideas to my project. I could have never asked for a better family at Bournemouth.

It is also my privilege to thank all my friends at the MA digital effects and MA 3D animation labs for their constant support and love. Special mentions to the people who filled out the survey and aided to the valuable research in this report.

I would like to thank my parents and my brother for being my pillars of strength every single day to help me go through this entire course with ease. I would also like to thank my best friend Sanjay for helping me out with all my needs and for his constant support. Finally, I would like to thank God for blessing me with opportunity to learn and grow as a person and helping me complete my course.

Table of Contents

List of Figures

List of Tables

1. Introduction

Fracturing is a phenomenon that is observed every day in our lives. We drop and object that is of a breakable material we fracture it. If we throw something at an object where the strength overpowers the strain of the impact object it leads to fracturing. Overall fracturing is a frequently occurring event in our lives.

Fracturing is defined as a process of splitting of a material over a region of lowest strain or at point of impact. The effect of fracturing in the entertainment industry has been extensively used. Films like Avengers Endgame [2019], Avenger's Age of Ultron [2015], Divergent [2014] and Snow white and the Huntsman [2012] have all demonstrated destruction of various materials to achieve almost accurate catastrophic events or to demonstrate fantasy. Nevertheless, in the world of computer graphics research behind various fracturing techniques are booming concepts.

Fig 1: Still from Avengers: Age of Ultron 2015

There are various types of techniques when it comes to fracturing techniques in leading visual effects software. Some of these include Volume based fracturing, Surface based fracturing and Dynamic fracturing. Volume based and Surface based fracturing are two pre-fractured methods whereas, dynamic fracturing uses an approach with concepts of physics based.

In this paper I will be investigating the pre-fractured techniques namely Volumetric based approach and Surface Scattering based technique. The two experiments will each be tested on concrete and wood. I will also be experimenting fracturing techniques on glass. The experiment on glass will be more inclined to a pattern generation tests. So, the tests on glass will consist of two surface scattering approaches. One will be the paint based method and another a texture based method.

In this paper I aim to conduct a survey to determine a very user centric approach to tackle fracturing in various materials. I will be further proving my research with visual representation and testimonies from various students of the NCCA. With this I will study what pre-fractured technique is most likely to be used and what makes Voronoi fracturing one of the most sought-after methods used in the visual effects industry.

The software that I will be using to conduct these experiments is Houdini 17.5.

2. Background Research

One of the most common approach based on physics to deal with deformations and fracturing techniques is by using the Finite Element method. The Finite Element Method or the FEM is a numerical based technique used to solve deformation problems using Partial Differential Equations (PDE) or functional minimalization. Finite elements are an assembly of finite elements. Node Values of a physical field is sought after to determine approximating functions. A continuous physical problem is transformed into a discretized finite element problem with unknown node values.

The first step to compute a finite element solution involves dividing a solution to a region of finite elements. The finite element mesh consists of several arrays, main of which are nodal coordinates and element connectivity's.[1]

Various fields of the element are interpolated using various interpolation functions. The interpolation functions are represented as polynomials. The polynomial degree is dependent on the number of nodes assigned to it. Further on the matric equation needs to be established to relate the unknown values to the nodal parameters. To come up with a global equation all the elemental equations must be assembled for discretization. Boundary conditions are imposed at this stage.[1].

The global equation of the finite element is always sparse, symmetric and discrete. The solutions require direct and iterative methods. This is how various mechanical problems of stresses, strains and displacements are solved.[1]

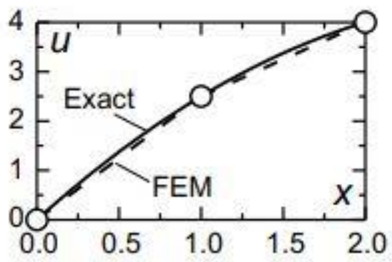

Fig2: Graph of comparison from the link

The finite element solver in Houdini is used extensively in rigid body dynamics. The solver studies the stress that an object experiences and either bends or breaks the object based on calculated values. The finite element solver is also used for volume preservation in soft body dynamics. The solver uses the above explained process of splitting finite number of elements based on the approximation of physics of continuous materials. In solid objects tetrahedrons are used as elements. In cloth objects, triangles and quadrangles are used to determine finite elements. Orientation of the tetrahedrons have very little influence when compared to the overall shape [SIDEFX].

The process of finite element method was first considered to be implemented as a part of this thesis. Although it was too extensive the topic needed to be deviated from the original content to fit the relevant time constraints.

3. Technical Background Research

3.1 Houdini

Houdini is one of the most prominent software's used in the film industries today. It helps in the work of major areas of production like modelling, dynamics, animation, rendering, compositing, volumetric and plugin development. Houdini has a nodal workflow making it a very procedural based network system to implement various a series of operators. The process of wrapping them up into digital assets give users advantage to port their work across various platforms. Various times new approaches that are more efficient and not complex can be implemented in place of existing operators to give more freedom for customisation.

One of the most prominent areas of Houdini is the rigid body dynamics. All collisions and simulation of motion in hard, solid objects are simulated using the Houdini Rigid Body Solver node.

Rigid Body Objects are divided into active objects and passive objects. Forces and collisions affect active objects. Objects that are not affected by forces of an active object and do not move are passive objects.

3.2 Concrete

A composite material like concrete is made from a rocky material mixed with a binder. Aggregate is made up of large amounts of material like coarse stones like limestone or granite and is blended with sand. The most commonly used binders in the concrete industry involve Portland cement or asphalt. There are various types of concrete some high strength, stamped concrete that is used mainly in architectural buildings and high performance concrete.

Concrete fracture is known to mainly occur when there is some sort of stress and deterioration that has been taking place over a period. The stable crack propagation depends on the amount of strain applied on the concrete. On application of load the concrete becomes unstable releasing the strain energy and self-propagate cracks until it completely breaks apart.

Fig3: References for broken rock

3.3 Wood

It is the structural tissue of trees and shrubs. It consists of cellulose, which makes it withstand strain and resists compression. Over the years wood has moved forward from simply existing in nature to being incorporated in our everyday lives. From every furniture we use to every fuel that drives a lot of machines, wood has made its way to being one of the most prominent components. Density of wood is determined by various characteristics. Age, diameter, height, radial trunk growth is some of the few factors that will greatly determine the wood density.

Fracturing of wood or wood- based materials undergoes two stages. The first one is crack initialization and the second is crack propagation due to the dissipation of energy. When a crack is initialized in wood it creates a micro crack in large numbers. But as propagation begins these micro cracks join to for major macro cracks. With increase in crack lengths bridging becomes weaker and the cracks finally take place.[3]

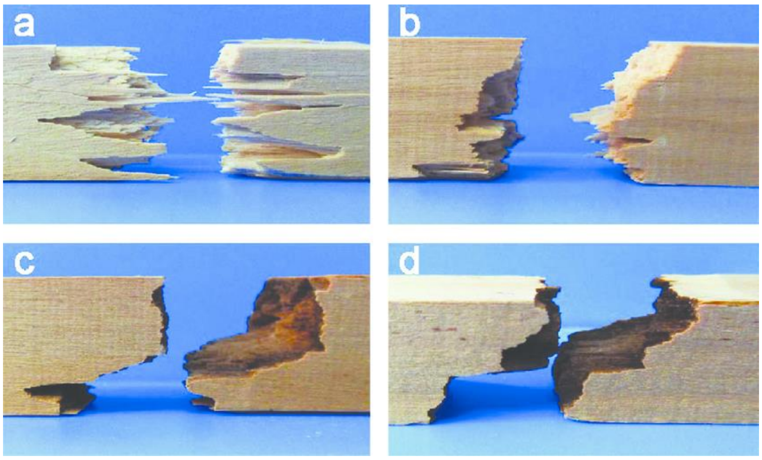

Fig 4: Reference for types of broken wood

3.4 Glass

The composition of glass involves the fusion of silica, sodium borosilicate and led oxide. Glass is one of the materials that tend to have different physical properties when breaking due to its distinctive physical properties. There are three main types of glass in the industry. Annealed, tempered and laminated impact resistant glass. Most of these glass materials look the same at first glance but on shattering the fracture pattern gives out what the actual type of glass it is. [4]

Fig 5: Reference for glass fracture

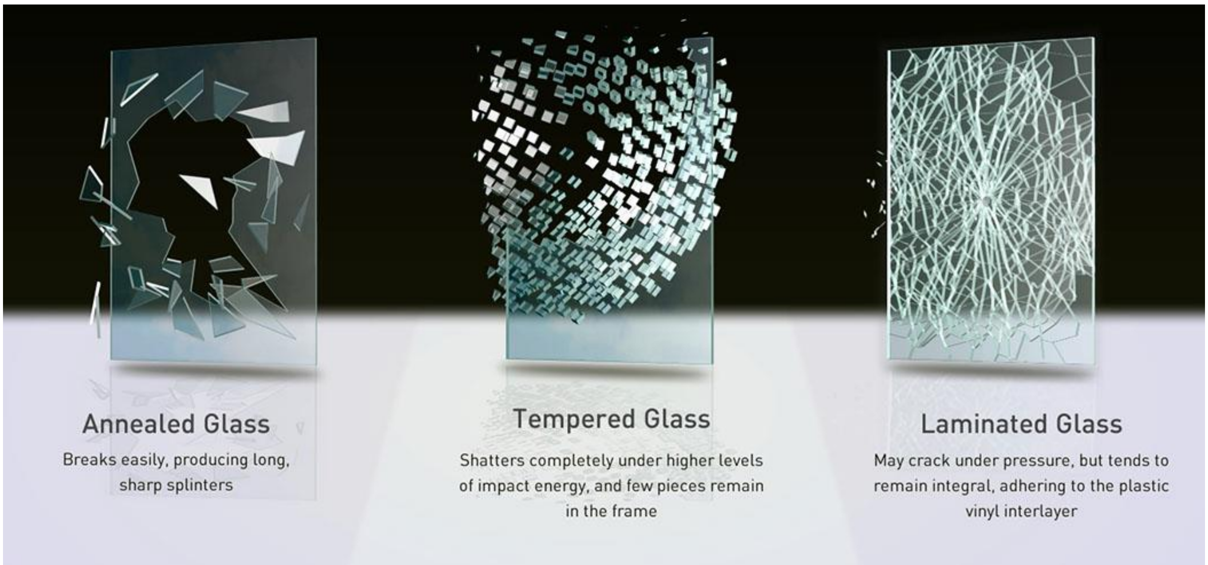

Fig 6: Types of Fracturing in glass

Fracturing of annealed glass demonstrates one of the most recognizable shatter patterns. This is achieved by cooling down the glass and allowing the internal pressure of the heating process to be slowly released, increasing the overall strength and durability of the glass.[4]

Annealed glass is much stronger than tempered glass and it is a safer choice in breaking because it forms granular diced pieces.[4] Laminated glass is widely recognized due to its spiderweb glass fracturing. It consists of compressed glass with laminations and its ability to stop the impact of shattering makes it one of the most suitable glasses to have. Laminated glasses are widely recognized at security buildings, airline windows etc.

The area affected by impact is called the strain. Stress is the force that causes the deformation. When stress is greater than strain a fracture, it begins to deform or break depending on the resistance. Glass being brittle shatters easily. Glass fractures from a little amount of stress energy. Once the stress is released glass cracks. [4] Glass is said to crack about 1,458 meters per second 3262 miles an hour.

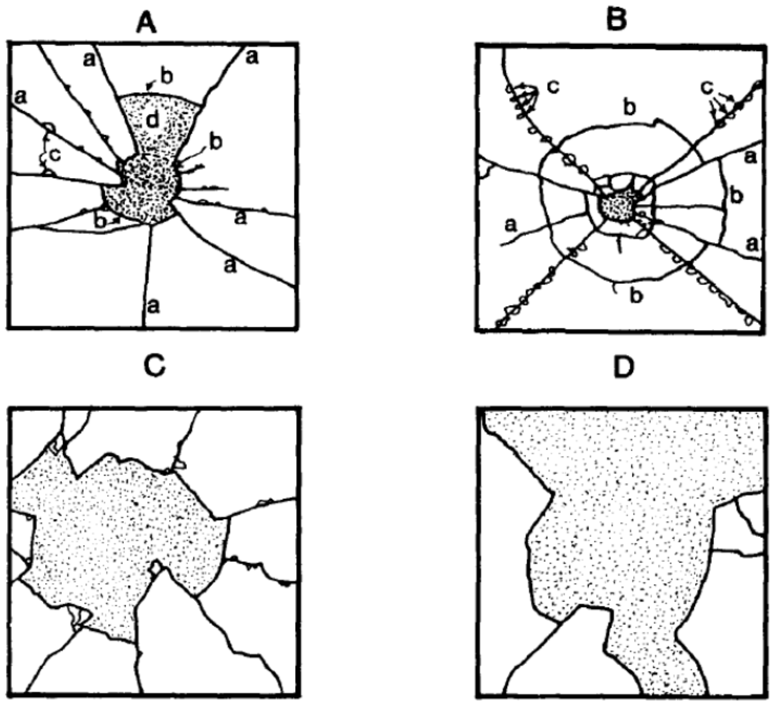

Fig. 2. Examples of broken panes showing the following features: A, 6 mm plain glass (1.0 × 1.0 m) radial cracks; B, 6 mm wired glass (1.0 × 1.0 m) circumferential cracks; C, 4 mm patterned glass (0.25 × 0.25 m) regions of surface flaking and D, 4 mm plain glass (1.0 × 1.0 m) hole resulting from large pieces dropping vertically down. Note: the scale of C is different from the other three diagrams.

Fig 7: Scientific diagrams for various types of glass fracture

3.5 Algorithmic solutions to fracture various materials

The most commonly used approach in VFX is the Voronoi algorithm Voronoi algorithm is a space division algorithm that divides a 3D mesh into various polygons. In the Voronoi node of Houdini when a seed point is added they compute the points closest to one and other and divide the mesh accordingly. There are many ways to manipulate a fracture pattern in Houdini. Changing the seed amount and adding randomisation to the points is one way. Another way is by scattering points along the volume of the object. Another way is by using textures. Simulating fracture patterns along this way creates an effective and an almost accurate way for multiple objects for different materials but in a more time efficient way. However, the glass material works a little differently. Simulating a glass fracture requires manipulating the simulations extremely. Tweaking the pattern sometimes creates a very different glass pattern in real life.

3.6 Voronoi Fracture

Voronoi diagram is defined as a partition of space based on the distance between points in that spaces' subset. The point will be predefined before the Voronoi pattern takes place and these sites are called seeds.

One of the most common methods to generate a Voronoi pattern is by Delaunay triangulation. Each point consists of a dual graph for the Voronoi construction. The vertices of the diagram are determined by the circumcentre of the Delaunay triangles.

Synthesizing a Voronoi pattern on the input mesh is one of the first steps to generating a crack. This process involves a point set that acts as the sees and a triangulation process for each polygon. This looped process creates a cluster of smaller polygons that cover the original mesh. On carrying out this process independently a Voronoi pattern is formed all over the polygon. The Voronoi pattern formed consists mostly of convex shapes that spread out from each vertex. [5] These shapes are approximated into a convex hull for a piece of visual mesh. To generate a Voronoi based convex hull the polygonal mesh is considered as an input based on Voronoi division of space. During offline mesh decomposition convex must be re-fitted to enclose a visual mesh. A detailed mesh consists of a large number of faces and vertices. The main requirements to satisfy the convex decomposition process is to enclose all the vertices of the visual mesh. It should not overlap and other convex surfaces of the compound. Connected convex must share co-planar faces. The main procedures that are undertaken are creating convex hull and updating a convex hull. This idea is a key to have each vertex store volume it would have to the current vertex. One of the implementations further discussed in this paper is just the simple generation of Voronoi pattern. That is bounded by a bounding box of the mesh. The algorithm converges to the true convex hull as the number of included vertices approaches the polygon vertices ad the volume approximation converges to the volume of the convex hull from the above.[6]

4. Implementation

In order to conduct a survey, it was decided that I implement two experiments against each material. For concrete and wood, two methods namely the surface scattering based Voronoi fracture and Volume based Voronoi fracture is implemented. Although for glass I decided to implement two techniques under surface scattering, one being a paint based method and another was surface based method.

4.1 Concrete

In order to achieve surface based scattering of concrete we first need a model or a filled polygon. The surface of the polygon will set the based for how the points will be scattered. With the help of basic scattering points are assigned to the surface of the mesh. The points that are added on become the seed for the generation of fracture patterns on the surface. The original polygon mesh and scatter points undergo Voronoi decomposition to form a cluster of fractured polygons. These pre-fractured polygons are then passed integrated with rigid body dynamics to create fracture simulations. The constraint part of the geometry helps the polygon glue together and fracture on impact giving it a more natural feel.

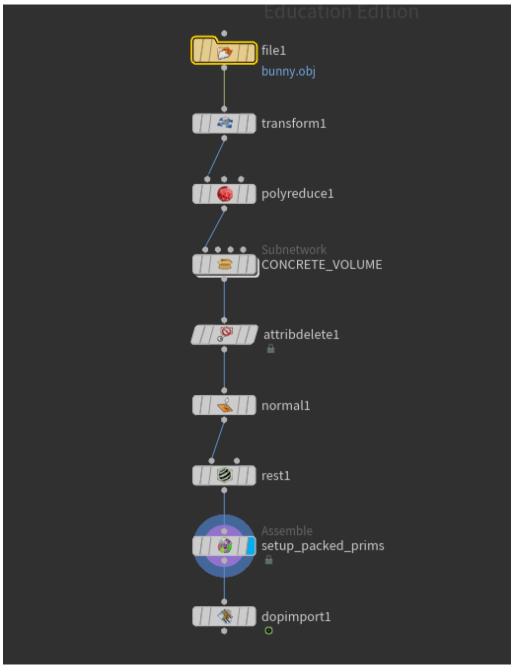

Fig 8: Concrete Volume test network

Another method that was used to experiment the scattering of concrete was Volume based fracturing where a solid polygon was taken as an input. An implicit function was used to create shell at a fixed offset from the original surface. Points are all scattered to the interior of the shell volume. All the points set inside the volume of the polygon are initialized as the seed to the fracture step. The original mesh and the scattered points are then passed to perform the Voronoi decomposition method. This method creates a cluster of internal polygons since the points were scattered internally.

In order to create the internal noise for concrete, the polygon was made to save the initial position. Then the entire polygon was distorted with a certain amount of noise. The polygon is then restored to the original position and remeshed. Once the fracture is applied and the internal surfaces are created the blast node was applied to sperate the internal and external groups. Deform attributes are created and the internal node is set to one for maximum distortion and the deform attribute outside is set to zero for minimum distortion. An attribute transfer is added to create a blend between the maximum and minimum distortions. The internal surfaces are then distorted once more using the deform blend parameter along with the maximum and minimum amplitude to give a shift value. This shift value sets the amplitude for the noise which is then added with the initial position to give the final output position. This creates the internal rigid effect. This same approach was used for the implementation of the surface scattering approach. There were issues where it distorted only the edges of the broken pieces.

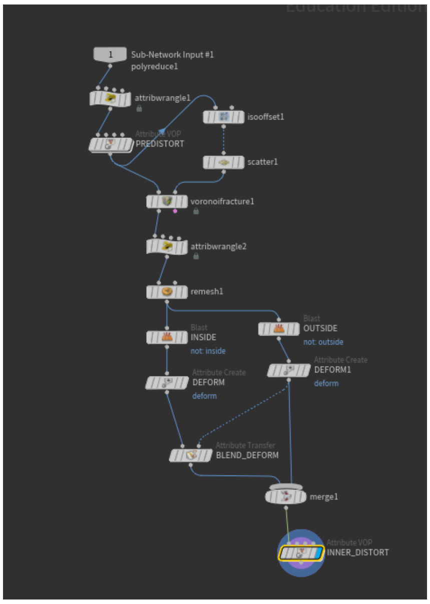

Fig 9: Node network to create internal noise

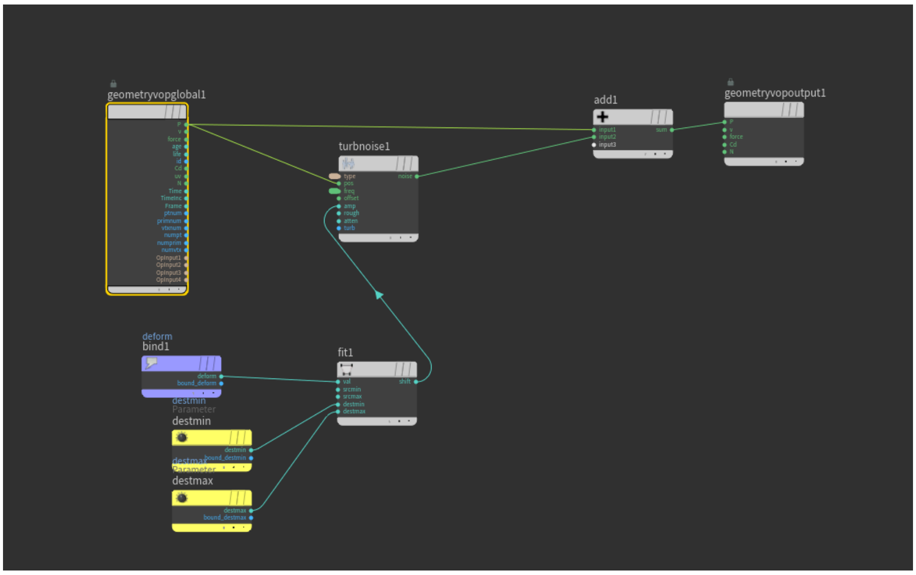

Fig 10: VOP that creates the noise

These polygons are then made as a rigid body and are set to be impacted by sphere to demonstrate a realistic fracture.

4.2 Wood

The tests involving wood were again divided into two methods. One was surface based Voronoi fracture and another was volume based Voronoi fracture.

To create surface based Voronoi wood fracture, a polygon is taken as the input geometry. Wood behaves very differently for fracturing. In order to achieve the almost natural aesthetic for wood, fracture points need assigned only at impact points to the geometry. These points will be the basis for setting fracture points across the wood. The original input mesh will be a box that is transformed at a specific axis to mimic wood. Points are scattered across the surface to a bare minimum. These scattered act as seeds to split the polygon box into fragments. The polygon is then grouped based on its interior surfaces. These groups are made to create an interior physical wood like fracture. For each interior group, noise is added to all the points of the interior face in the axis along which the wood has been scaled. The noisy axis is merged with the other two axis and then assigned as new position. This creates a sharp geometry along the interior surfaces of the wood. The fractured polygon is then passed to be a rigid body model and fall on impact and fracture.

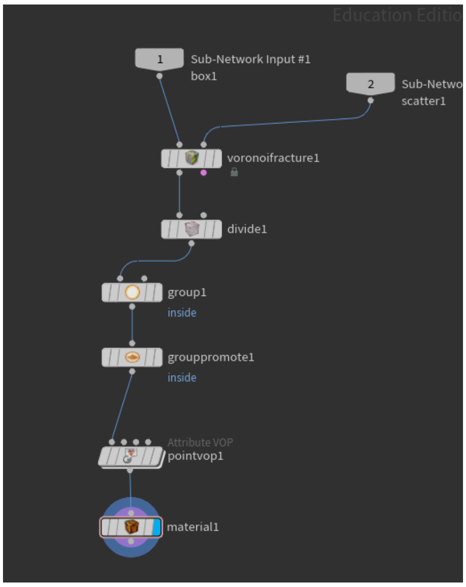

Fig 11: Surface scatter wood fracture node structure

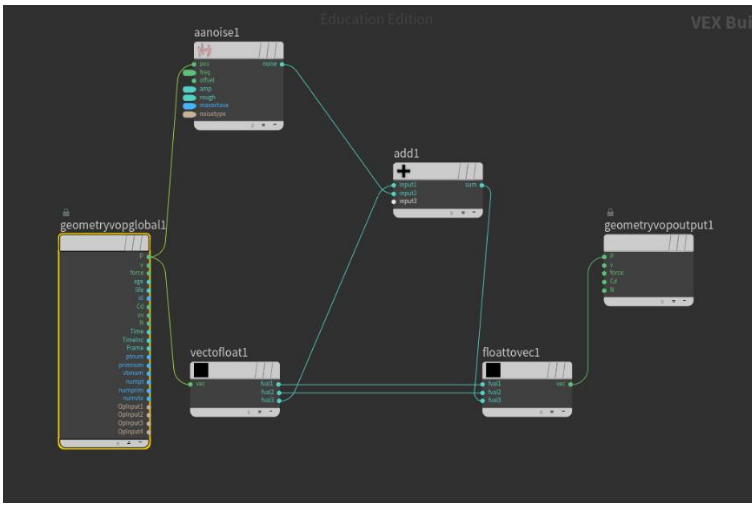

Fig 12: Noise to generate splinters

The volume based Voronoi fracture is implemented by using a box polygon as an input geometry. This input geometry is then converted to a volumetric shell. Where a single point is scattered onto its density. This point is then used to generate a closed area of many packed scatter points which will be used to influence only a single area of a polygon. These newly generated points and the original box polygon are considered as input to be fractured. The resulting fracture creates a cluster of polygons at close proximity to each other. It also results in internal fractured polygons. The fractured mesh is then subjected to being scaled across a specific axis. This makes the smaller cluster of patters elongated to create splinter like behaviour of wood and resultant breaking. The wood is then assigned as a rigid body which is then set to drop and break on impact. One of the positives of this approach is that this fracture creates debris to make it look very fractured. This technique will be accurate to create simulations of low wood quality.

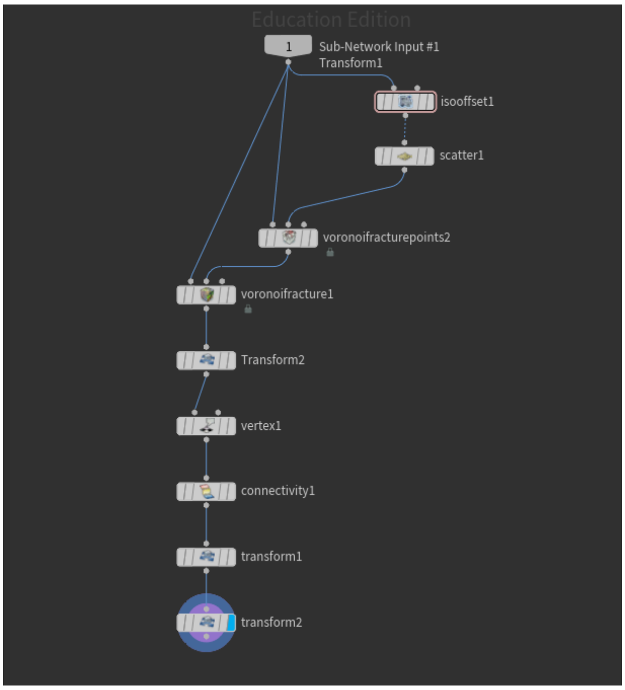

Fig 13: Wood volume test node structure

4.3 Glass

For the simulation of fracture on glass a different approach was taken. I decided to implement only a surface based Voronoi fracture approach but by using two different techniques. One is using the paint based method and another one uses textures to help scatter the points. This was used to help study if Voronoi can be used for glass fracturing as well.

For the paint based approach the input mesh was a box polygon that was scaled across different axis to make it look like a glass slab. Concentric circles are painted across the surface to add specific shape to mimic what a concentric glass fracture would look like. These concentric circles are assigned to points. The points set up are set as seed to be decomposed by the Voronoi pattern. The pattern that emerges is a radial pattern of smaller polygon segments. These segments are transformed to a rigid body object which is then fractured on impact. There is an impact sphere generated dynamically to grow as the sphere impacts to depict a dynamic fracture like the glass.

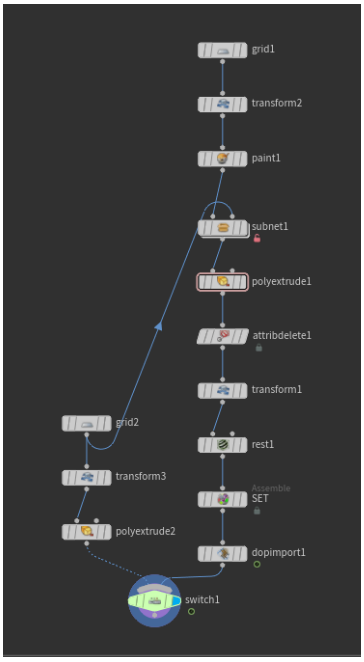

Fig 14: Paint based glass fracture

Another method used to achieve an accurate glass fracture is using a texture. An image based texture of already existing glass fracture is used to demonstrate the cracks. The images are converted to a monochromatic texture that depicts cracks at the white colour and the rest of the surface as black. This method takes in a box scaled to look like a glass slab as an input. The generated monochrome texture is set as an input to the trace function. Once traced, points are scattered along the white lines on the group to create a distinct pattern. The points are then scaled appropriately to mimic a glass fracture, like annealed glass. The transformed box polygon and the scattered point from the trace are fed as inputs to the Voronoi fracture which creates dense fractures around the area that has more scatter points and larger fractures for the points spread out. Once this is achieved the resulting pre-fractured polygon is set as a rigid body object to be fractured. The object is then set to be simulated as a fracture that grows radially on impact. This is achieved by creating a bounding sphere that is set at a size a little bigger than the impact sphere and then is animated to grow radially as the impact takes place.

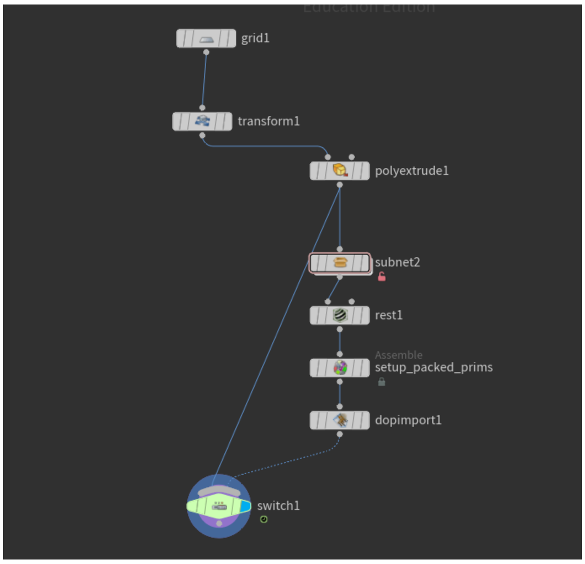

Fig 15: Texture based glass fracture node network

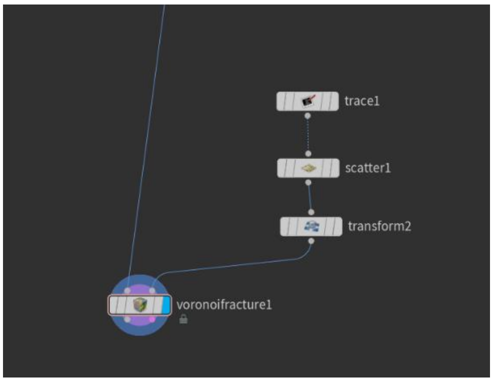

Fig 16: Tracing of the fracture pattern

A survey was conducted to determine what they believed to be an optimum technique. This survey measured criteria that would benefit them primarily as users and achieving a decent result. The parameters that were used to determine this were determined by freedom that the user has, the effort that the user needs to put in, the accuracy and realness of the fracture and performance of the simulation.

4.4 Implementing convex decomposition

Further into this research I decided to implement Voronoi fracture using C++ and OpenGL. The main implementation goes along the lines of the report by Muller et al. The code consists of loading an input geometry in the form of objects into the code. It then adds pre-fractured points onto the geometry and splits the polygon in to depict a convex decomposition of the object.

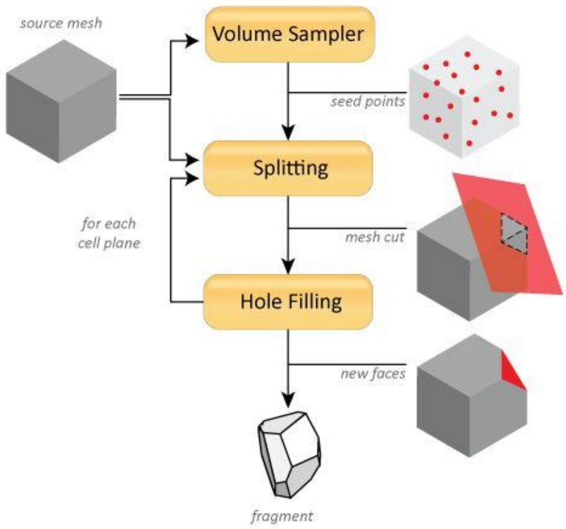

Fig 17: Voronoi based convex Decomposition

In the implementation the main scene loads an object file. It is then initialized, and the scene matrix is initialized. Once the scene Matrix is loaded the simulation begins by updating the pre-set fracture points. On initiating the simulation, the fracture pattern loads in the points by accessing the object name. The fracture pattern is added and is set for computation.

The Edge class focuses on creating the edge structures by triangulating the faces and creating pairs of inner and outer vertex data. Once faces are partitioned the centre of mass is calculated by calculating the average of the centroid. The edge class also declares functions to add, remove and update the fracture points. This class also initializes a function to orderly pair up edge pairs. In order to process the render data, the edges computed are accessed and made into ordered lists of vertices and normal. Once the vertex list and normal list are computed.

The bounding box class has an extension function to the ngl::BBox class which compute the max and min for every x,y and z which is then called to update and draw the bounding box for the scene.

The cut mesh class it initializes the edge class and iterates a colour counter over the entire loop for colour changes for every mesh that is split. The number of planes to be split are iterated and the centre point of the plane is initialized. It then loops over the mesh and gathers all the data needed to cut the mesh which is stored in the Polygon structure. The polygons are then clipped. The clipping starts by determining the equation of a plane and for each vertex if the corresponding side is less than zero it is retained or clipped. Every convex shape is sorted and triangulated and gathered for clipping again. Once cut the vertices are initialized into a new face structure and edge pairs are initialized again for the new list of faces obtained.

The Planes class initialize the necessary functions to compute the plane required for clipping the mesh. It is a header that initializes the Voronoi points, computes normal and calculates the point centre.

The compound class calculates the fracture pattern. It iterates over the number of pre-fractured points assigned and makes fracture pairs. These fracture pairs are then normalized, and the centre point is calculated in order to initialize the plane needed for clipping.

The utility class initializes the pre-fractured points and the colours for the colour counter that would be used and assigned as and when a new clipping is formed.

The main implementation of this code was to deeper understand the volume decomposition that various material undergoes before collision. The bounding box acts as convex hulls for each convex or concave mesh computed.

5. Results

Results of the implementation of the of the various pre-fracturing techniques

5.1 Concrete

5.1.1 Surface based

The surface scattering for concrete (Fig 18) achieved desirable results although it gave out a very sliced effect. This gives a very unnatural representation on how concrete is fractured in real life. It was a difficulty to implement internal noise onto the internal surfaces of concrete.

Fig 18: Result of surface based scattering for concrete

5.1.2 Volume based

Volume scattering of concrete (Fig 19) demonstrated more solid results and realism to concrete. The result demonstrated chunks of concrete due to internal Voronoi decomposition due to the internal points. This method also depicted internal noise more accurately to give a rough concrete effect.

Fig 19: Result of volume based scattering for concrete

5.2 Wood

5.2.1 Surface based

Surface scattering of wood (Fig 20) gave a partially realistic fracture but a good interior detail. The implementation made it look like the impact was made of hard wood, but hard wood doesn't break that way. The interior detail gives a good effect of the wood but the edges on where the wood fractures doesn't always have the effect. It gives very clean cuts, and this always isn't accurate.

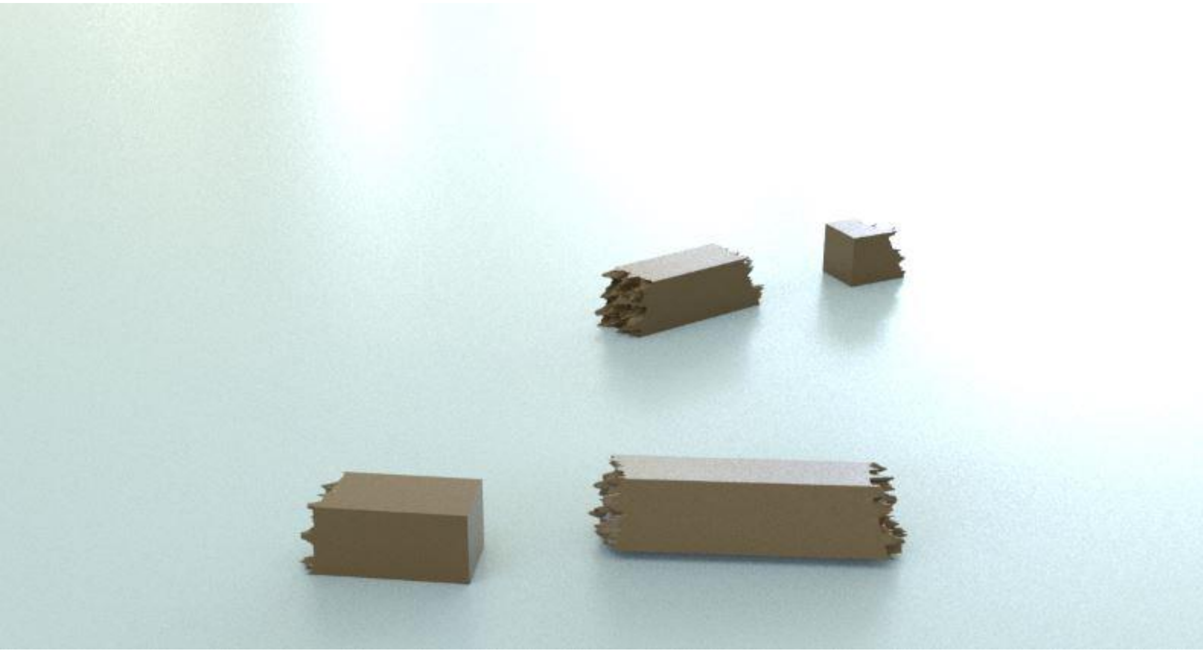

Fig 20: Result of surface based scattering for wood

5.2.2 Volume based

Volume based scattering yielded accurate results. It demonstrates the look of soft wood. It shows the beginning point of impact, wood fracture propagation and formation of debris and splinters. The interior detail generated is accurate. Overall the entire look of this fracture technique was very close to accurate.

Fig 21: Result of volume based scattering for wood

5.3 Glass

5.3.1 Paint based

Paint based glass pre-fracturing technique gave out a concentric result but not accurate enough to realism. It is a good effort to mimic for a tempered glass approach, but it gives out a more uniform triangular result which is not entirely how glass cracks

Fig 22: Result of Paint based scattering for glass

5.3.2 Texture based

Texture based glass pre-fracturing gave the closest result to an annealed glass fracture. The main characteristic of annealed glass is the formation of large shards of glass on impact. The texture allowed us to scatter points exactly along the lines of the cracks. This resulted in a crack pattern identical to the texture incorporated.

Fig 23: Result of Texture based scattering for glass

6. Survey Observation

6.1 Results for the concrete fracturing technique.

The survey for concrete roughly compared volume based and surface based scattering to determine the accuracy, user freedom, effort and performance. The simulations were compared to real aesthetics of concrete fracture. It also compared the freedom and effort it took to build these simulations.

The first parameter calculated accuracy (Fig24) where each technique was compared to determine which technique leans towards a more accurate fracture pattern in concrete in Houdini. The survey determines on an average 81.8% believe that volume based pre-fracturing technique performs better than the surface based pre-fracturing technique 15.2%. This the case since the surface based technique tends to have a more sliced way of fracturing whereas the volume based pre-fracturing creates internal Voronoi polygons due to internal fracturing. Although these results are subjected to change depending on where it is used in the industry volumetric based pre-fracturing is a more preferred approach to achieve a more accurate fracturing of concrete.

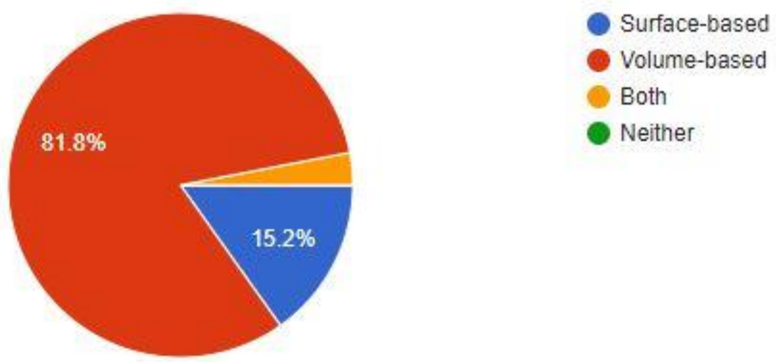

Fig 24: Accuracy of concrete fracture

The second parameter used to demonstrate a more effective approach to fracturing concrete was effort (Fig 25). It is a parameter used to benefit the user on which fracturing technique is more time consuming to create in Houdini. Numbers from the survey show that 69.7% prefer the volume based approach to be more time consuming. This is because surface based scattering dynamically scatters more points over a defined volume unlike the surface based scattering that takes more time to predefine points on where it needs to be scattered. There are a percentage of people that be.

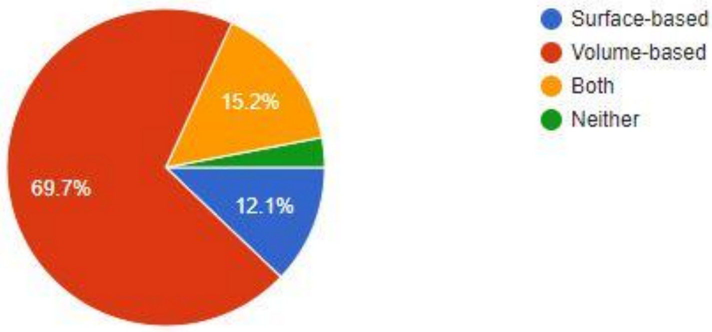

Fig 25: Effort parameter for concrete fracture

The third parameter that was used to demonstrate a more effective approach to fracturing concrete fracturing was the freedom (Fig 26) that the user has in determining where to scatter points on concrete. Many of the users had varied opinions to this answer. Since both the types of fracturing can have freedom based on the where and how volume and surface based scattering can look like. In this survey majority of the people believe that surface based scattering 39.4% gives more freedom for the user to have control over the fracture whereas 33.3% believe that volume scattering gives way for more freedom. 18.2% say that both the types of fracturing have equal freedom and it depends on the situation.

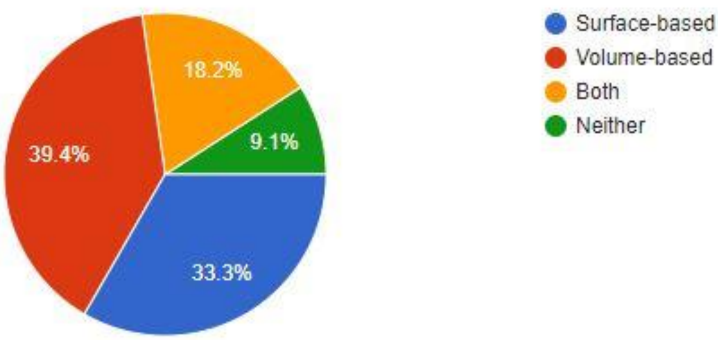

Fig 26: Freedom of concrete fracture

The last parameter that was used to determine concrete fracturing was the simulation speed (Fig 27). This parameter is used to determine the time taken by the Houdini interface to simulate a fracturing technique. Majority of people believe that surface based scattering 69.7% would simulate faster since points are already predefined onto the polygon. However, 24.2% feel that volume based scattering would take a longer time to simulate. This is because Houdini takes a longer time to generate a volume from a polygon the more. For larger objects with a lot of detail and high resolution converting them into detailed volumes will result in high simulation times to scatter points.

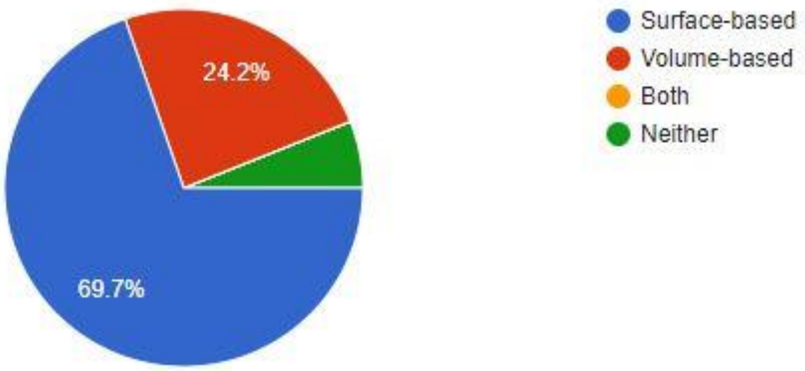

Fig 27: Simulation time for concrete fracture

6.2 Survey of pre-fracturing techniques on wood

The survey for wood compared volume based pre-fracturing and surface based pre-fracturing. These techniques were compared based on accuracy, freedom, effort and performance. The accuracy and performance are the parameters used to determine the user's opinion on software efficiency whereas the freedom and effort were used to user defined preference.

The first parameter that was surveyed was the accuracy, this parameter is used to determine whether the wood fracture of both volume based surface pre-fracture and surface based pre-fracture are accurate enough to reality (Fig 28). In this survey 69.7% of the people believe that volume based scattering is more accurate than surface based scattering 24.2%. With people who don't agree with surface pre-fracturing method, pointing out that surface scattering looks more like a block of iron. I do agree that volume based scattering is more accurate than surface based scattering since the internal cracks helps in simulating debris. Volume based scattering also creates more accurate splinters and irregularities on the remaining blocks.

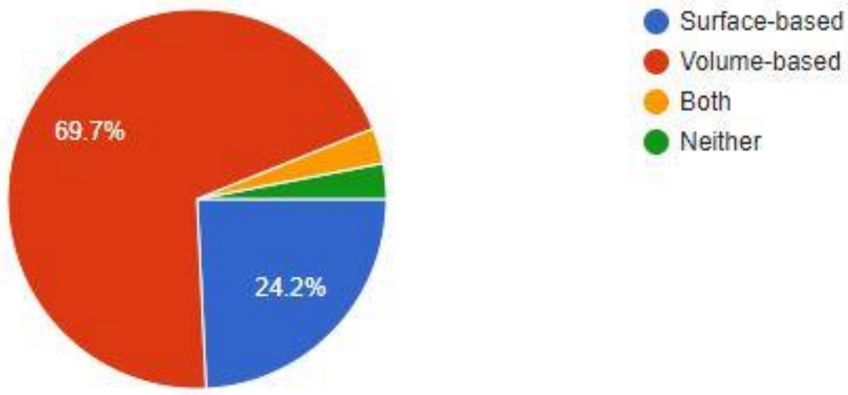

Fig 28: Accuracy of wood fracture

The second parameter that is used to compare the two methods is the time or the effort to build the entire system. 87.9% say that it would take them longer to make a volume based wood fracture than the 6.1% who say that scattering points and adding noise would take up lesser the amount of time.

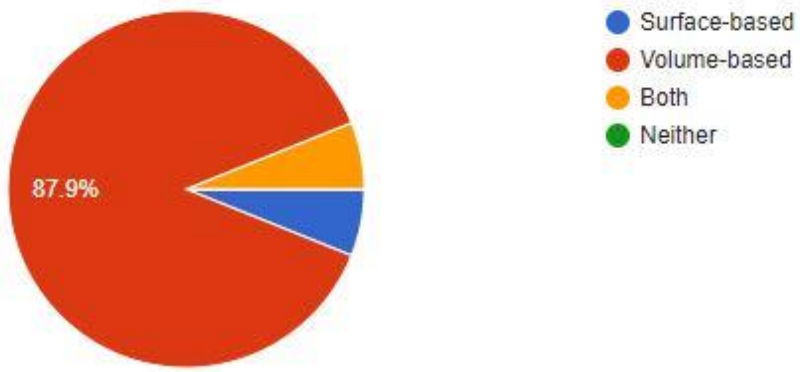

Fig 29: Effort parameter for wood fracture

The third parameter that is used to compare the volume pre-fracturing and surface pre-fracturing is the freedom 75.8% believe that volume based scattering gives more freedom since everything is directional across a specific axis. 12.1% say that surface based scattering is gives a better result. Although both techniques yield a satisfactory result the volume based scattering gives a lot more freedom to simulate better fractured wood.

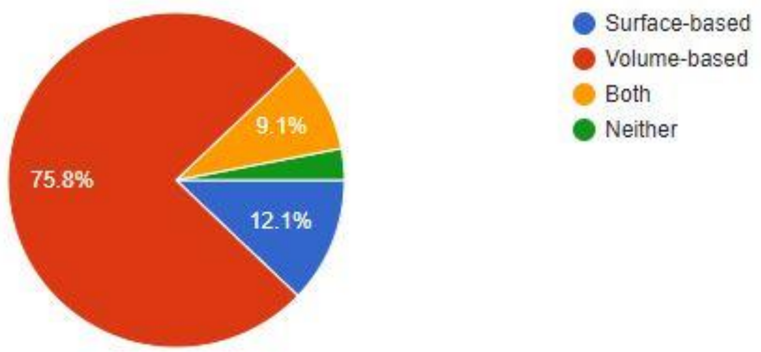

Fig 30: Freedom of wood fracture

The fourth parameter used to compare the two pre fracturing techniques of wood is the simulation time. This parameter is used to determine the amount of time a user would need to wait till the entire asset simulates. Out of 33 people who took up this survey 84.8% feel that surface based scattering would take up lesser the amount of time. While making my test for both the techniques it turns out that not only my approaches did take a significant amount of time to simulate but the volume based scattering took much longer than the surface based approach.

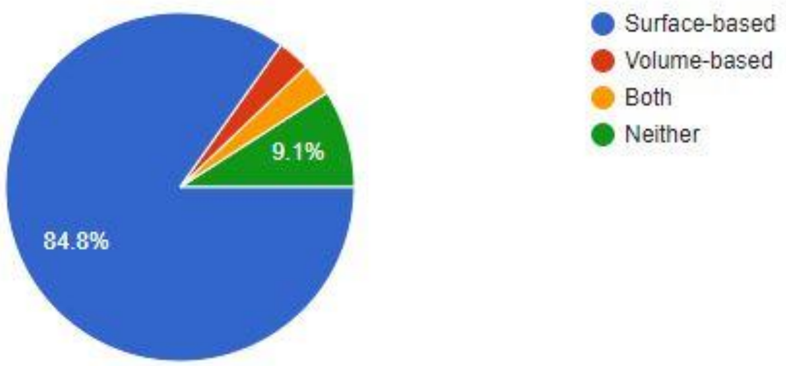

Fig 31: Simulation time of wood fracture

6.3 Survey of surface pre-fracturing techniques for glass

The survey for glass consisted of different approach. I decided to implement only surface based scattering on a glass slab to determine the accuracy of pre-fracturing techniques for glass. The techniques used were an approach of painting scatter points and the other was a texture based approach. These simulations were also tested against for parameters like accuracy, effort, freedom and simulation time in the form of a questionnaire answered by my colleagues.

The first parameter that is analysed is accuracy where both the approach of painting scatter points and using a texture for surface pre-fracture were compared to demonstrate how close they are to reality. 84.8% of the people found the approach to using textures yields a more accurate result. 9.1% of the people found that painting the surface scatter points yields a better result. 6.1% of the people felt that both demonstrate an accurate result. Although to simulate actual breaking of glass. The number one parameter to be concerned is with the type of material that needs to be implemented. I believe that texture-based approach is a great way to give annealed glass type properties and the approach by painting and converting them to scattered point demonstrates a better way to depict cracks formed by concentric glass. Although the scattering of points by painting can achieve glass like scattering with major tweaking, I don't necessarily find it a very effective approach since the scatter points do not create extremely pleasing results.

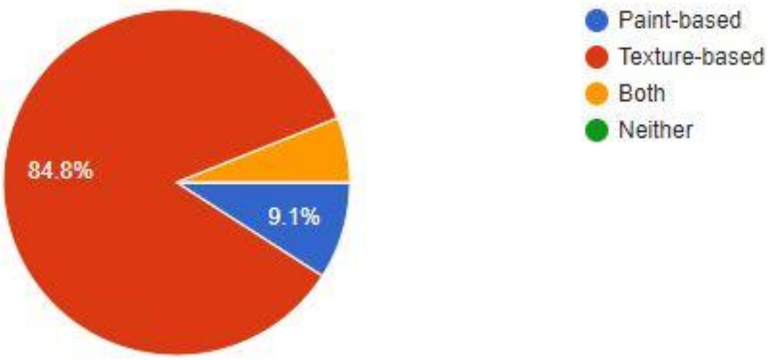

Fig 32: Accuracy of glass fracture

The second parameter that was examined between the two glass fracturing techniques was time or the effort it would take for a user to build the entire network. It was found that 51.5% said that it would take more time to paint and build an accurate class crack. On the other hand, 36.4% believe that it would take them less time to make a pre-fractured system used to crack glass. While building the node network for both systems it was found that using textures to create crack used lesser the amount of time to make and gave out a much more efficient result. 12.1% of the people believe that it would be time consuming for both the methods to be implemented.

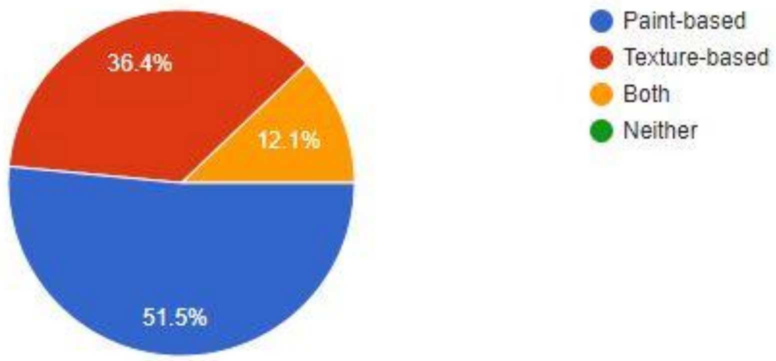

Fig 33: Effort parameter of glass fracture

The third parameter that is used to compare the two methods is Houdini's ability to have more freedom to create both the techniques. 49.9% feel that paint and scatter method give more freedom to the user since the user can paint on the polygon to scatter points wherever they wish to do it. 31.3% say that scattering based on texture yields better results. 21.9% agree that both give equal amounts of freedom. And that is the case since both the techniques allows the user to define points as per the user's interest.

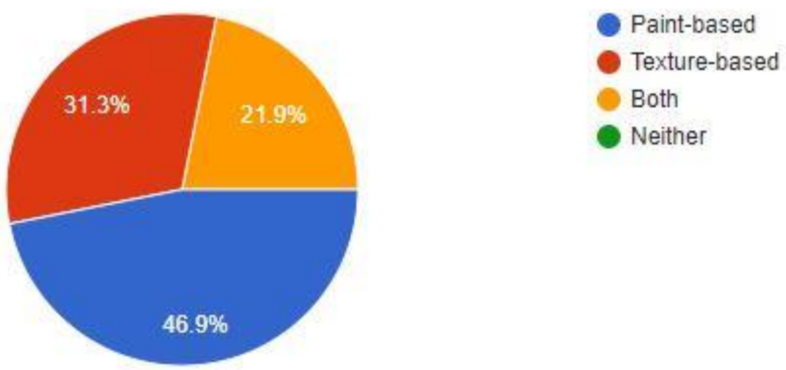

Fig 34: Freedom parameter of glass fracture

The fourth parameter used to analyse the efficiency of both the techniques is the simulation time. 42.4% of the people believe that pre-fracturing using the method of painting points simulates faster. 36.4% believe that pre-fracturing using textures simulates faster. 15.2% have confirmed that both the surface scatter approach and surface texture-based approach has fast simulation speeds. On conducting the experiment, it did take longer to simulate the paint based method because of the numerous scatter points and it took time to generate Voronoi decomposition for them.

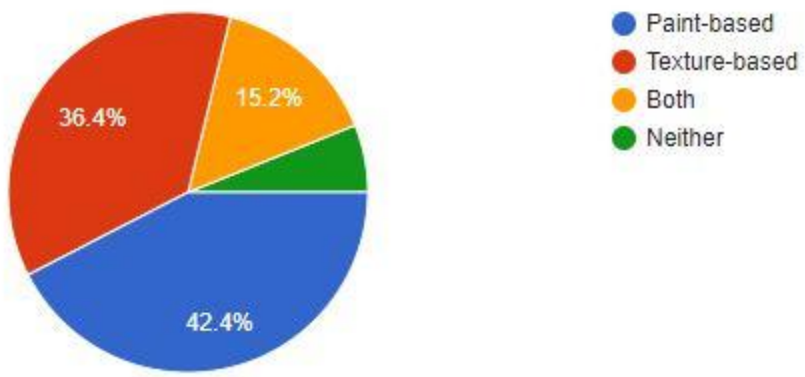

Fig 35: Simulation time of glass fracture

7. General results of the survey

7.1 Survey results for concrete

Material: Concrete	Surface Based	Volume Based
Accuracy	--	✓
Time	--	✓
Freedom	✓	✓
Simulation time	✓	--

Table 1: Survey results for concrete fracture

For the general analysis of concrete, it is found that most of the users achieve accuracy, save time with implementing the volume based approach even if the environment gets more complex. As for freedom, people say that freedom is a very debatable topic since you can achieve certain amounts of freedom in both the volume based and surface based technique. As for the simulation time volume scattering takes time to generate scatter points on the surface which instantly increases the simulation time. Overall despite the time and effort people still do agree that volume based scattering achieves better results. Since simulation times by software are heavy anyway achieving realism is of importance.

7.2 Survey results for wood

Material: Wood	Surface Based	Volume Based
Accuracy	--	✓
Time	--	✓
Freedom	--	✓
Simulation time	✓	--

Table 2: Survey results for wood fracture

For the wood tests, most of the users confirm that volume based scattering produces a better result with the object looking like it underwent wood propagation and chipping. Users who said that surface scattering is better because of the internal chipping gives better accuracy. Although people who disagree with surface scattering say that the fracture pattern looks heavier and could be applied only to very rare cases. Users also say that it would take a lot of effort to create a volume-based fracturing method. Users also said that volume based scattering provides more freedom. However, most of the users also say that the surface based scattering takes up lesser the amount of simulation time. Which is true. Generating scatter points over volume is automatically simulated by Houdini and this drastically increases simulation time. But, with respect to accuracy to realism and the scientific logic of breaking of wood, volumetric scattering produces better results and achieves the closest accuracy.

7.3 Survey results for Glass

Material: Wood	Paint Based	Texture Based
Accuracy	--	✓
Time	✓	--
Freedom	✓	--
Simulation time	✓	✓

Table 3: Survey results for Glass

In the case of glass, only surface based test was conducted using the paint based approach and texture based to experiment on pattern accuracy. For accuracy the cracking the glass material using a texture to scatter points, produced an accurate annealed glass fracture to which most of the users agree. People who took up the survey also agreed that using paint to scatter points was more time consuming. Surprisingly users say that paint based method gives more freedom than texture based method. Whereas both the methods have relatively fast simulation times. Overall achieving the look of glass is tricky. At present, achieving annealed glass fracture is the closest we can get to achieving a realistic glass fracture.

8. Conclusion and Future Work

This dissertation taught me different ways a user could approach the same problem. It gave me a detailed understanding of how various methods to the same problem can weigh very drastically in their outcomes. This project helped me understand Houdini better by the implementing different methods to fracture the same materials. Attempting to try to implement various other concepts also not only gave me the scientific knowledge but the bases to attempt them as bigger projects in the future. In the future it is also of interest that I would like to dig more deeper into various glass fracturing pattern generation. Although with the existence of the new node RBD material fracture in Houdini's 17 versions I would hopefully like to attempt to implement a technique that focuses mainly on the effects and generation of glass fractures.

Lastly, I would once again like to thank my parents, faculty and friends from the NCCA for being a part of my life and helping me grow this past one year. It has been an eventful experience and I hope to carry over and implement all my learnings in the next phase of my life.

9. References

1. *Vallin, L., 2017. The simulation of continuous casting of steel using finite element method*
2. *www.sidefx.com*
3. *Stanzl-Tschegg, S.E. and Navi, P., 2009. Fracture behaviour of wood and its composites. A review COST Action E35 2004–2008: wood machining–micromechanics and fracture. Holzforschung, 63(2), pp.139-149.*
4. *Monroe, D.C., 2014. A Materials Science Driven Pattern Generation Solution to Fracturing Computer-Generated Glass for Films and Games (Doctoral dissertation).*
5. *Raghavachary, S., 2002, July. Fracture generation on polygonal meshes using Voronoi polygons. In ACM SIGGRAPH 2002 conference abstracts and applications (pp. 187-187). ACM.*
6. *Müller, M., Chentanez, N. and Kim, T.Y., 2013. Real time dynamic fracture with volumetric approximate convex decompositions. ACM Transactions on Graphics (TOG), 32(4), p.115.*
7. *Su, J., Schroeder, C. and Fedkiw, R., 2009, August. Energy stability and fracture for frame rate rigid body simulations. In Proceedings of the 2009 ACM SIGGRAPH/Euro graphics Symposium on Computer Animation (pp. 155-164). ACM.*
8. *Terzopoulos, D. and Fleischer, K., 1988, August. Modelling inelastic deformation: viscoelasticity, plasticity, fracture. In ACM Siggraph Computer Graphics (Vol. 22, No. 4, pp. 269-278). ACM.*
9. *O'brien, J.F., Bargteil, A.W. and Hodgins, J.K., 2002, July. Graphical modeling and animation of ductile fracture. In ACM transactions on graphics (TOG) (Vol. 21, No. 3, pp. 291-294). ACM.*
10. *Bargteil, A.W. and Cohen, E., 2014. Animation of deformable bodies with quadratic, Bézier finite elements. ACM Transactions on Graphics (TOG), 33(3), p.27.*
11. *Nealen, A., Müller, M., Keiser, R., Boxerman, E. and Carlson, M., 2006, December. Physically based deformable models in computer graphics. In Computer graphics forum (Vol. 25, No. 4, pp. 809-836). Oxford, UK: Blackwell Publishing Ltd.*
12. *Kaufmann, P., Martin, S., Botsch, M. and Gross, M., 2009. Flexible simulation of deformable models using discontinuous Galerkin FEM. Graphical Models, 71(4), pp.153-167.*
13. *Pfaff, T., Narain, R., De Joya, J.M. and O'Brien, J.F., 2014. Adaptive tearing and cracking of thin sheets. ACM Transactions on Graphics (TOG), 33(4), p.110.*
14. *Goffredo E. 2010. A tool for procedural destruction in Houdini shattering + dynamics. Masters. Bournemouth University.*
15. *Sharma H, 2013. Soft body deformation dynamics based on Shape matching. Masters. Bournemouth University*