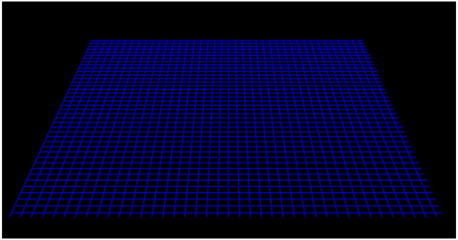
# Simulating Persisting Water Droplets on Water Surface

# Water Surface Definition

- I have defined the water surface as plane in the 3D space, composedof a set of N * N connected vertices, where the point P(i, j, k) has 3 identifiers, i (X-coordinate), j (Y-coordinate), and k (Z-coordinate).
- This will result in a grid-like shape in the 3D space.



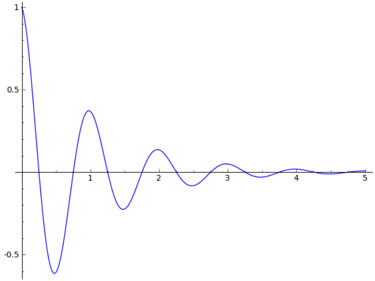
# Vertex definition

- For simulation purposes, a vertexs X and Y coordinates wont changes through out the whole process, because the water surfaces position wont change during the simulation, but the variable parameter in this case will be the height of every vertex in the 3D space as an effect of water waves and height propagation in a function of time

# Vertex definition

- For simulation purposes, a vertexs X and Y coordinates wont changes through out the whole process, because the water surfaces position wont change during the simulation, but the variable parameter in this case will be the height of every vertex in the 3D space as an effect of water waves and height propagation in a function of time
- Resulting in a Damped Sine Wave graph as the following:

$$y(t) = e^{-t}.\cos(2\pi t)$$



# Three dimensional geometric transformations

- When a water droplet falls in the center of the water surface, the height of this vertex will change as an effect to the droplets weight, and as a consequence, all the neighbours vertices heights will change also in a function of the Damped Sine Wave function.

# Three dimensional geometric transformations

- When a water droplet falls in the center of the water surface, the height of this vertex will change as an effect to the droplets weight, and as a consequence, all the neighbours vertices heights will change also in a function of the Damped Sine Wave function.
- $$P(x, y, x) \Rightarrow P'(x', y', z')$$
$$P'(x', y', z') = T(tx, ty, tz).P(x, y, z)$$
$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} . \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

# Connecting water surface vertices

- In order to let the water surface feel more realistic and visual, I had to connect all thevertices with each other, to reach a grid like shape on a 2D plane in the 3D space. And thatlets us see the water surface is always connected even if some vertices Z-axis value variesin effect of a droplet fall on the surface.

# Connecting water surface vertices

- In order to compute every vertexs height in a given state during the wave simulation process, I had to develop an algorithm to do this. I have used Dynamic Programming to compute a vertexs height according to the neighbouring vertices, where the Z-axis value of a vertex in a given state is defined by:

## Connecting water surface vertices

- In order to compute every vertexs height in a given state during the wave simulation process, I had to develop an algorithm to do this. I have used Dynamic Programming to compute a vertexs height according to the neighbouring vertices, where the Z-axis value of a vertex in a given state is defined by:

$$D(x, y) = \frac{\text{Sum of neighbouring droplets}}{2} - D(x, y)$$

Where

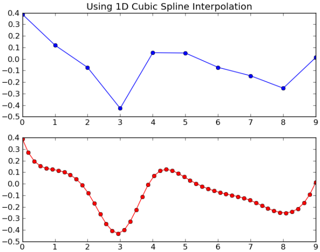
$$D \Rightarrow \text{a droplet on the water surface (vertex)}$$

$$x \Rightarrow \text{X-coordinate of the point}$$

$$y \Rightarrow \text{Y-coordinate of the point}$$

# Connecting water surface vertices

- In mathematics, a spline is a special function defined piece wise by polynomials. In interpolating problems, spline interpolation is often preferred to polynomial interpolation because it yields similar results, even when using low degree polynomials.



# First Trial

- In my first trial, I tried to draw the water surface as a set of unconnected vertices in the 3D space, but the simulation was not realistic, it just showed some points move in a 3D space

# Second Trial

- In order to make a more realistic simulation, I had to introduce connections between vertices, the edges were straight lines, I connected every set of horizontal N vertices with each other, then connected every set of vertical N vertices with each other, which resulted in a grid view in the 3D space.
- This trial was not very successful as there was no smoothness in the simulation scene,the connections between vertices was very sharp, which did not simulate the water waves in the right way.

# Last Trial

- My last trial was the last, I solved the problem of edges sharpness using splines connecting between all the surface vertices.This resulted in the best simulation scene could be reached, but still not realistic.
- The problem was that all vertices were connected using splines, but all the spline lines pass through the vertices, which introduced smoothness but still not realistic as using B-splines,where the splines does not mandatory pass through every vertex in the surface plane.

Presented by Yahya Alaa Massoud

Thank you.