Problem Chosen	2020 MCM/ICM Summary Sheet	Team Control Number
REPLACE ME WITH PROBELM		REPLACE ME WITH TEAM NUMBER

Summary

After building a spectacular sand castle, we always think to ourselves: "how wonderful would that be if the castle could stand forever!" Unfortunately, waves rise and ebb, washing everything away. However, there are certain useful strategies we can adopt to preserve our sand castle for as long as possible, which is the topic of our article. Based on **Mohr-Coulomb Criterion** and **Cellular Automata**, we construct a model to identify the best 3-dimensional shape of our sand castle foundation under waves and tides. First, we break down the shape-identifying process and establish two corresponding sub-models to determine the side view shape and top view shape.

Side View Shape We adopt **Mohr-Coulomb** Criterion to calculate the critical angle of the side slope and demonstrate why it turns out to be the optimal shape.

Top View Shape **Cellular Automata** is used to identify the top view shape of the foundation. We develop a set of rules concerning the force of the wave and water content in different sand layers, and simulate the erosion process of our sand castle foundation. To solve for the best shape, we use two sets of enumeration rules:

1) Begin from the square shape, then

2) Begin by selecting key points and perform **Interpolation Methods**, from the simplest **Lagrange Interpolation** to **Cubic Spline Interpolation**.

The effect of rain is simulated by adjusting several

On determining the best sand-to-water proportion, we

In the following section, we list six strategies that might make our sand castle last longer to further demonstrate the influence of *salinity of seawater*, *compactibility of sandpile*, etc.

In the end, we perform sensitivity analysis on our model and discuss the strengths and weaknesses of it. Our model features high flexibility as it is able to simulate the erosion process of sand castle foundation of any shape.

Kingdom Built on a Pile of Sand: Slow and Steady

Hello, dear readers of *Fun in the sun*!

As Leonardo da Vinci says : Simplicity is the ultimate sophistication." Maybe you find it difficult to build Pyramids or Sphinx in deserts as Egyptians did. However, you will be soon in your element creating a long-lasting sandcastles on the seashore If you try some of our important tricks for you.

When it comes to summer, building sand castles is definitely on the must-do list of beach goers. However, you may know how vulnerable they are if you have built a sand castle once. A gentle wave can cause a damaging collapse, and even if your sand castle fights its way through the hitting of waves, it will be eventually washed away by the waves and nothing will be left. What a pity! Is there a way to help our sand castles last longer? Well, recently, a team of MCM try to construct a mathematical model to determine an ideal Strategy to build a long-lasting sandcastle and finally tackled this problem.

#1: modify the shape of foundation Well, If you're building from memory, then first envision your castle. Just like cars are streamlined to minimize resistance of the air, a well-designed shape of your foundation can evidently prevent waves to flush away the sand and erode your sandcastles. Maybe a shape that have a smaller contact surface or enables to reduce the impact of waves will make your castle's life longer. You may try our team's new design shown in the figure to turn a pile of sand into a spectacular sandcastle!

#2: proper sand-to-water proportion An optimal sand-to-water mixture proportion makes your sandcastle support its own weight against rough waves. Just a little bit of water enables liquid bridge forms between grains, which enables your sandcastle stand as a whole. Too much water, however, will destabilize the material and it will be easily washed away. In fact, our suggestion for you about the optimum liquid volume fraction is about.

#3: use terrain to your advantage As Mencius says:"The time isn't as important as the terrain", so make full use of the terrain could make your work better. For example, choosing a beach near a less salty ocean may contribute to building a more concrete sand castle. What's more, build your sandcastle at a proper distance from shore will obviously reduce the damage caused by waves, and building a moat along the castle can help foundation survive the rising tide.

#4: Better material, Longer life If you are devoted to try all methods as you can do, then you should take material into consideration. Finer sand is a bonus because by constructing our sand castle foundation using finer sand. In addition, you may try some special material, such as glue, concrete to make your sandcastle tougher.

Sand is an ephemeral medium that requires a builder to approach the work with equanimity. So have a fun in summer and enjoy building your own kingdom above the sandcastle!

Contents

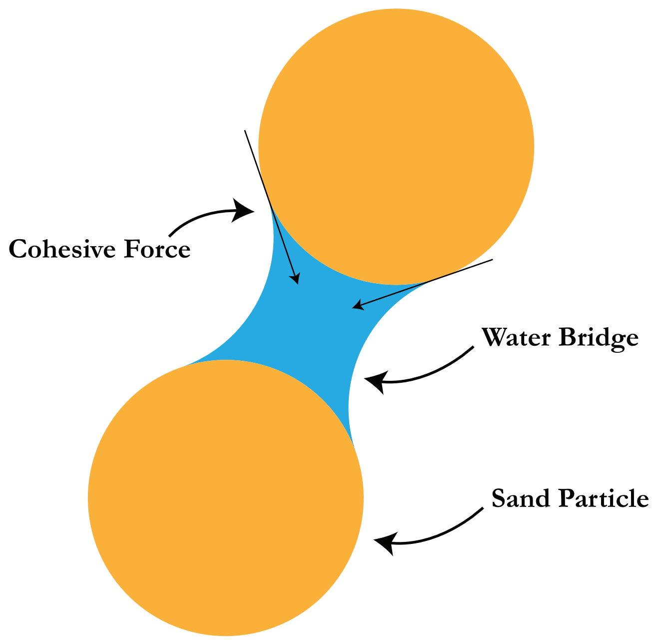


Figure 1: Water Bridge Between Sand Particle

1 Introduction

1.1 Problem Background

Sunshine, clear blue sea and golden color sand always seem to leave people in a happy state of mind. And a beach is where these three are combined, drawing people all around towards it. Sand, the granular matter formed by constant brushing of flowing water, however, can react with water in a different way, despite the fact that people refer to it as non-stable or unreliable. On a beach, where the already formed granular sand and the rise and fall of sea wave lies together, a new buff can be added to our flowing friend, a wetted state.

Magically but not randomly, sand gets sticky when combined with water, due to the most obvious physical theorem: surface tension and atmosphere pressure. From previous people's work, we've know that this buff comes from the water bridge formed between sand particles, which can significantly cluster together during the increasing water-sand portion[1]–[3]. Kudrolli and Arshad has visualized the bridge between sand like Figure 1.

And that's the magic that glue our favorite sand castle together, which is one of the most entertainment for enthusiastic beach goers. However, being built near the water that melts mud and our wet sand, all sand castles have to face the fate that they'll g et too wet to hold its own weight and the impact from sea waves. That's because the water bridge has another property of clustering together[3]. When you throw a pile of sand into water, they behave just as when they're completely dry, melting down like fluid. That is, sand castles lasts for only a period of time. Thus beach castle builder might want to make their sand castle last longer than those build arbitrarily than others, which is also the purpose of this article.

1.2 Our Work

Normally people would sculpt their kingdom from a pile of tedious wet sand. To simplify the model and grasp main threads, we'll focus our research on this single, nondescriptive mound of wetted sand. In this article, we develop a method for determining the most long-lasting three dimensional shape for a tedious sand castle, which will be later referred to as *sand castle foundation*.

Our model addresses the problem of constructing a three dimensional structure by projection, which makes the construction of individual part of our model easy to implement. Firstly we determine the optimal *side view* of our *sand castle foundation* by analysing and extracting information from similar research on wet granular material[2]. Then we construct the best *top view* of the *sand castle foundation* by constructing a **Cellular Automata**.

Side Slope Determination You've probably seen questions about the slope of a pile of dry sand in junior high school practice books. In our research, we address the problem about dry sand as a normal junior high student would. However, the analysis of wet sand requires some technique beyond low level physics. In our research, we establish this model using **Mohr-Coulomb Criterion**.

Top View Determination Our model analyses the impact of sea wave flow from a top view, or a *slice* of the *sand castle foundation*. Then we can combine the sliced impaction result together with the *side slope* analysis to form a possible 3D object. Inside our model we construct a **Cellular Automata** since the shape of the sand slice edge isn't as obvious the side slope may be, that is, it's merely possible to construct continuous function on the shape.

Other Factors It's worth noting that the interaction between sand and sea wave is far a lot more complex than direct force impact or sand-to-water mixture proportion change. Some other factors include: gravity of sea water that submerge the sand underneath, influx of weaker sea wave and the gravity of water inside the wetted sand. During the assembly of two views, we'll address these other factors that may influence a *sand castle foundation*'s life span.

2 Assumptions & Nomenclature

2.1 Assumptions

Several assumptions are made in order to simplify our model.

1) The weather condition is suitable for sbuilding sand castles. Namely, no huge waves occur.

2) The sand castle foundation is set where the maximum height of wave does not exceed a certain value, which we set as 0.2m for convenience, when the tide rises.

3) Our optimization goal is to acquire the largest top size while minimizing the lost of sand. Intuitively, if the side of the foundation is built as a gentle slope, the possibility of collapsing will be drastically reduced. However, with a fixed volume of sand, such a foundation will not allow any sophisticated carving.

4) Sand grains are regarded as tiny spheres. In our foundation, these grains are of the same size and are closely packed.

5) The top of the foundation is a flat platform whose surface is horizontal.

6) The side further from the sea has the same slope as the side facing the sea, as when waves recede, its

2.2 Nomenclature

Table 1: States of Sand

Symbol	Meaning
τ	the shear stress on a plane
τ_f	the shear stress on the failure plane
μ	the internal friction coefficient
σ	the normal compressible stress on a plane
σ_f	the normal compressible stree on the failure plane
ρ	the density of sand
θ_c	the critical angle of a sandpile
D	the height from the top of the sand pile
P	the pressure[]
Γ	the surface tension of water
s_A	the adhesive stress
f_A	the adhesive force
V	the total amount of fluid present per particle contact
R	the radius of particle
$c_{ijk}(t)$	the cell mixed with sand and water to form the sandcastle in time t
$s_{ijk}(t)$	the amount of sand in the cell in time t
$w_{ijk}(t)$	the amount of water in c_{ijk} in time t
$p_{ijk}(t)$	the proportion of sand-to-water mixture c_{ijk} in time t
$S(t)$	the amount of sand in the sandcastle in time t
$W(t)$	the amount of water in the sandcastle in time t
$P(t)$	the average proportion of sand-to-water mixture c_{ijk} in time t
$F(t)$	the erosion-osmosis common effect of the sea water in time t
$T(t)$	the tangent of the sand surface

3 Modeling Under Waves and Tides

The 3-dimensional shape constructing problem is divided into two sub-problems. We first establish the model using the **Mohr-Coulomb Criterion** to decide the shape of the slope, then construct another model with a modified version of **Cellular Automata** to determine the best shape viewed from the top.

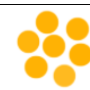
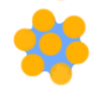
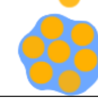
3.1 Shape of the Slope: Mohr-Coulomb Criterion

We begin by determining the side shape of the sand castle foundation. Before Approaching the problem, we will briefly address the property of sand as a granular media.

In our assumptions, sand particles are considered as identical tiny spheres. If we zoom in to observe a pile of wet sand, there are the so-called liquid bridges formed between sand particles.

Various water contents produce different liquid bridge distributions, which will influence the properties of sand.

Table 2: States of Sand

Liquid Content	State	Description	Schematic
No	Dry	No cohesive force exists	
Small	Pendular	Liquid bridges start to form	
Middle	Funicular	Liquid bridges and liquid-filled pores coexist	
Almost saturated	capillary	Almost all pores are filled with the liquid	
More	Slurry	No cohesive force exists	

When the wave gets into contact with the foundation, the surface area is in slurry state and there exists no cohesive interaction between the particles, which makes it very hard, if not impossible, to prevent sand loss in this process. Nevertheless, collapses after the wave resides can cause more harm to the foundation, which can be avoided by alternating the shape.

When the wave gets into contact with the foundation, the surface area is in slurry state and there exists no cohesive interaction between the particles, which makes it very hard, if not impossible, to prevent sand loss. Nevertheless, collapses after the wave resides can cause more harm to the foundation, which can be avoided by alternating the shape.

For dry sand, the failure criterion is given in terms of the shear stress τ, the normal compressible stress σ and the internal friction μ as

$$\tau > \mu\sigma$$

This is simply the friction formula with different notations. For wet sand, we consider a sandpile with a normal adhesive stress s_A across every plane, in addition to the stress caused by weight. The equation(1) is then modified as

$$\tau > \mu(\sigma + s_A)$$

This is the so-called **Mohr-Coulumb criterion**. The stress resulting from the weight above the plane is shown in figure(2). Denote τ_f and μ_f as the shear stress and normal compressible stress at the failure plane, it is obvious from the schematic that they can be written as

$$\tau_f = \rho g D sin\theta_c \qquad and \qquad \sigma_f = \rho g D cos\theta_c$$

where θ_c is the critical angle, D is the height of the sandpile and ρ is the density of sand. Therefore, combine the equations above, to solve for θ_c is to solve the equation

$$\mu = tan\theta_c(D)\left(1 + \frac{s_A}{\rho g D cos\theta_c(D)}\right)$$

The only unknown factor is s_A, the adhesive stress across the plane. According to (Thomas C.H and Alex J.L -fix later)'s study, the value of s_A is determined by water content and there are three regimes as a function of the added-fluid volume. We now focus only on the state where the water content is close to saturation.

In this case, with water serves as lubricate, it makes sense to model sand particles as frictionless spheres. The pressure difference is then given by []

$$P = -\frac{\Gamma}{\sqrt{V/2\pi R}}$$

where Γ is the surface tension of the fluid, V is the total amount of fluid present per particle contact, and R the radius of the particle. According to the study of [fix later], the adhesive force in this state is given by

$$f_A = 2\pi\Gamma R$$

Note that (with certain conditions like distance between grains remains constant)the term V which denotes the volume of liquid per particle does not appear in equation(6).

We now focus on obtaining the value of s_A via f_A, whose value can be calculated with the equation above. Assume our foundation contains sand particles that are closely packed. In such a structure, we have $\phi_V = \sqrt{(2)}/6$, where ϕ_V stands for volume fraction which is defined as ratio of the volume of particles to the total volume. The average number of contacts per unit area will be $(3\phi_V/\pi R^2)$. With f_A representing cohesive force of a single liquid bridge, we can then write

$$s_A = \frac{f_A}{\sqrt{2}R^2}$$

Substitute the above result into Eq.(), we obtain the result

$$tan\theta_c = \mu + \frac{\sqrt{2}\pi\mu\Gamma}{R\rho g D} sec[tan^{-1}(\mu)]$$

3.2 Top View Shape

We used a modified version of **cellular automata** to simulate the impact of tides and waves. There are several factors that are taken into consideration:

3.3 Assumptions

1) We consider only the slice of the *sand castle foundation*, which is a two-dimensional plane. The sea wave is considered to come from above, which mean it impacts our *sand castle foundation* downwards.

2) Only a part of the *sand castle foundation* is taken into consideration since the chaos caused by flowing water would be too hard to simulation. So we'll only consider the impact angle's influence on both sand loss rate and water osmosis rate.

Note that all values aren't exact corresponding to what there actual value in real world might be, since we've seldom needed to define unit in computation or trying to map variables to exact physical concept. Instead, during our simulation, we consider more about development efficiency and the asymptotic complexity of the process so as to get more accurate results in a limited amount of time on a limited computer.

Simulation based on discrete **cellular automata** takes abundant amount of time to finish, and tends to lose accuracy when processing under low resolution. Therefore we've added the `slow` boolean variable to make the automata behave as a robust system would. Do remember to turn this switch off if you're to run batches of simulation.

3.4 Nomenclature

We use `Python`, a script language to construct our **cellular automata**: *sand castle foundation* simulator, which provides ample 3rd party library to speed up development. The **cellular automata** is sealed in a class of python called `SandCastleSimulator2D`, which has the ability to:

1) Computing current sand surface slope, which is going to affect how sea wave is going to affect this small area of sand.

2) Reverse information about current simulation field, include the state of each individual sand cluster, the water-to-sand proportion of each cluster.

3) Simulate a small moment of wave impact, dropping sand and updating each cluster's *humidity*.

4) Comstructing a color map for this particular problem.

Table 3: Variable Nomenclature

Symblol	Meaning
`width`	the width of the simulation field along x axis. unit: sand cluster
`depth`	the depth of the simulation field along y axis. unit: sand cluster
`delta`	a value small enough for slope computation, determines the rate of sand property alteration per wave impact
`initial_edge`	the front edge of the *sand castle foundation*
`shear_rate`	how much the angle of impact influences the stay or leave of each cluster
`humidify_rate`	how much water can penetrate into sand through outer sand shell
`humidify_depth`	the reciprocal of water penetration decrease rate
`initial_humidity`	initial humidity of each sand cluster
`osmotic_rate`	similar as `humidify_rate`, affects sand cluster no matter the angle
`osmotic_depth`	similar as `humidify_depth`, affects sand cluster no matter the angle
`slow`	whether the model is running on high resolution mode (bigger than 500x500)
`osmosis`	whether osmosis is considered (penetration no matter the angle)

3.5 Implementation

Initialization During initialization, we convert function arguments to object attributes to make the usage of these variables easier. And we also initialization several factors depending on the user's choice, for example `self.slow` and `self.osmo`.

During the process of initializing `delta_humidity`, which describes how water can penetrate sand, we estimate the water penetration as an exponential function according to previous study on water-concrete penetration [4]. however, further study and closer examination using more accurate device is still required.

Listing 1: Implementation of `__init__`

```python
def __init__(self, width, depth, delta,initial_edge=None, shear_rate=1,humidify_rate=0.005, humidify_depth=1, initial_humidity=0.1,slow=True, osmosis=False):
    assert (width > (delta * 2)) # delta shouldn't be too small to compute derivation
    # initializing attributes from user input or default value
    self.delta = delta
    ...
    ...
    # The front of the sand castle: self.edge
    if initial_edge is None: # default front is a transformed cosine function
```

```python
            self.edge = depth - ((np.cos(np.linspace(-np.pi * 3, np.pi * 3, width)) + 1) * depth / 10 + depth / 2).round(0)
        else:
            self.edge = initial_edge
        self.edge = np.asarray([0 if x < 0 else (depth - 1 if x > depth - 1 else x) for x in self.edge], dtype=int)
        self.update_slope() # calculate slope according to edge shape (discrete differentiation)
        # life: whether a sand cluster is still attached to the main foundation
        self.life = np.array([[i >= self.edge[j] for j in range(width)] for i in range(depth)])
        # humidity: a representation of water-sand proportion, not necessarily an exact value
        self.humidity = np.full((depth, width), fill_value=initial_humidity)
        # Perform static osmosis if required
        if self.osmo:
            self.osmosis()
        # Update humidity according to sand cluster's existence
        self.drop_dead()
        # Delta humidity is initialized and approximated using an exponential function
        self.delta_humidity = np.flip(np.exp(np.linspace(-10 / self.humidify_depth, 0, self.depth))) * self.humidify_rate
        self.delta_humidity = np.tile(self.delta_humidity, self.width).reshape(self.width, self.depth)
        self.delta_humidity = np.transpose(self.delta_humidity)
        # Colorization: initializing color map for sand and water representation
        ...
        ...
```

Wave Impact The most essential part of this **cellular automata** simulator is the method `wave`, which simulate an **instant** when a *sand castle foundation* is being hit by wave. By **instant**, we mean the flow of sea water is considerable slow so as to be viewed as directly downward. And the complicated interaction between sand shape and water flow is also able to be neglected. During this instant, according to:

$$\Delta P = F \cdot t$$

We would also neglect the interaction between sand clusters as the momentum change δP approaches zero. This implies the sand can considered still during the **instant** of impact.

According to the most obvious mechanical analysis shown in Figure 2, the sea wave's impact is separated into two parts. One will take away a lot of sand, and the other will hit the *sand castle foundation* hard by penetrating into it, adding "humidity" along the way.

Here's a list of things to note about the implementation of `wave`

1) We approximately determine a sand to be melt down when:

$$\tau + \frac{1}{1 - W_i(t)} - 1 > 1$$

and

`self.slow` is `True` or $T_i(t) > 1$ (which can also be interpreted as $\cos(\arctan T_i(t)) > \frac{\sqrt{2}}{2}$)

2) When the resolution of the simulation is large enough, we're able to slow down the iteration by tweaking several places of the code to get a more accurate result.

Listing 2: Implementation of `wave`

```
def wave(self):
    cosine = abs(np.cos(self.slope))
    sine = abs(np.sin(self.slope))

    delta_shear = sine * self.shear_rate
    for k in range(self.delta * (1 if self.slow else 2)):
        for i in range(self.width):
            if ((cosine[i] > np.sqrt(2) / 2 or k < 1) or self.slow) and (
                    delta_shear[i] + 1 / (1 - self.humidity[self.edge[i], i])
                    ↪  - 1) > 1:
                self.edge[i] += 1 if self.edge[i] < self.depth - 1 else 0
                delta_shear[i] -= delta_shear[i] / self.delta
        self.drop_prick() # drop the pricks in the resulted edge
    if self.osmo:
        self.osmosis() # perform osmosis if required
    delta_humidity = self.delta_humidity * cosine

    #update humidity
    for i in range(self.width):
        self.humidity[self.edge[i]::, i] += delta_humidity[0:self.depth -
        ↪  self.edge[i], i]
    self.update_slope() # compute discrete differentiation on the updated
    ↪  edge
```

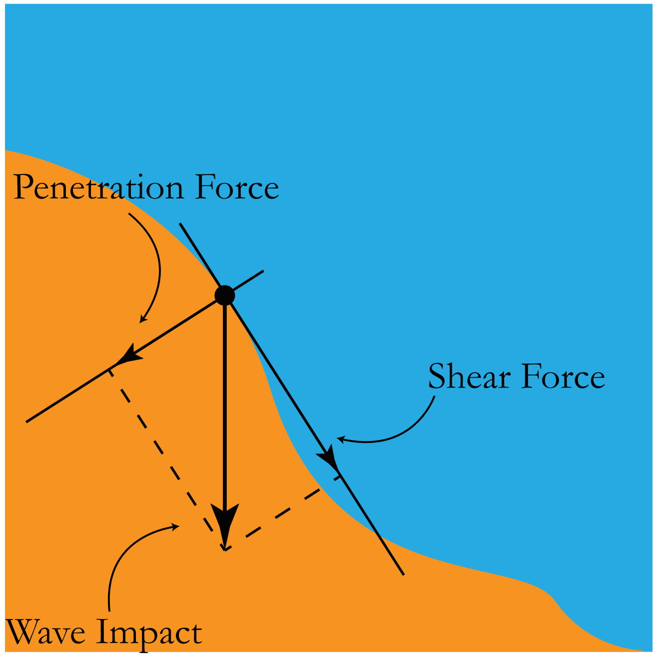


Figure 2: Mechanical Analysis

Osmosis By **static osmosis**, we mean the natural penetration of water through object, which has already been observed [1]. According to **Huygens–Fresnel principle** [5], we know that

"Every point on a wavefront is itself the source of spherical wavelets, and the secondary wavelets emanating from different points mutually interfere. The sum of these spherical wavelets forms the wavefront." as shown in Figure 3

Here in this question, we can consider the static osmosis phenomenon as the sum a series of point emanated secondary wavelets, which can significantly simplify the computation of the **normal** of the sand edge.

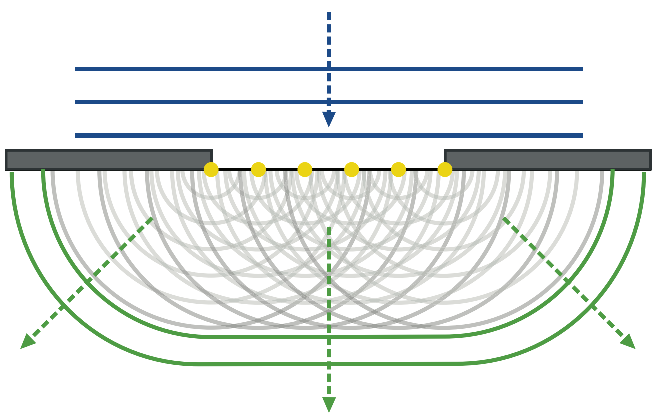


Figure 3: Huygens-Fresnel principle

However, during implementation, we still find the computational complexity is unacceptable for a quick, determinative simulation. Image if one simulation runs as long as a day, we might as well choose to observe the real world behavior of wet sand under wave impact. So during the process actual simulation, we choose not to consider static osmosis, treading accuracy and stability for efficiency. Of course, if the reader's got a CPU with a high enough single core performance, he/she might as well try to simulate the situation under complete consideration of static osmosis.

Other Methods Here we list other method used in the `SandCastleSimulator2D`.

Table 4: Method Explanation

Method	Usage
`drop_dead`	update `self.life` according to current front edge
`drop_prick`	smoothen current front edge
`update_slope`	compute the tangent (slope) of current edge, neglecting brink

3.6 Sample Simulation Result

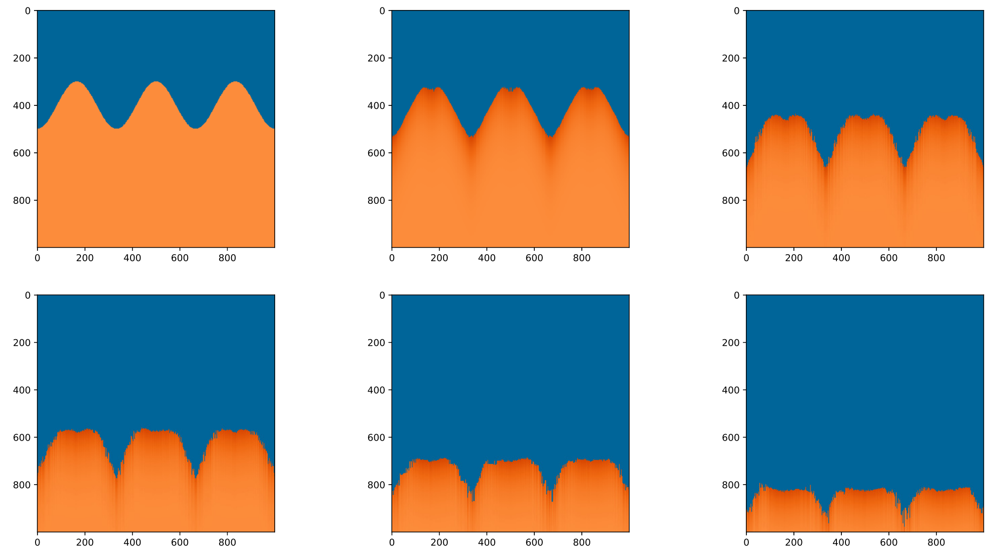
Here's a simulation result of the default configuration.

In Figure ??, we generate an output per hundred iteration with a setup like this:

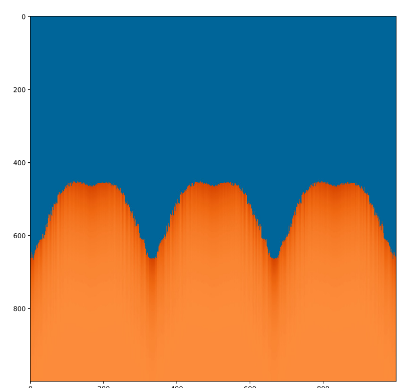
Listing 3: Generating Simulation Result

```python
from cell import SandCastleSimulator2D
sim = SandCastleSimulator2D(1000, 1000, 3, humidify_depth=1, shear_rate=0.75)
fig = plt.figure(figsize=(20, 20))
fig.add_subplot(231)
plt.imshow(sim.humidity, cmap=sim.sea_wet_sand)


for i in range(5):
    for _ in range(100):
        sim.wave()
    sim.update_life()
    sim.drop_dead()
    fig.add_subplot(230 + i + 2)
    plt.imshow(sim.humidity, cmap=sim.sea_wet_sand)
```



(a) Sand Cluster Loss and Water Penetration



(b) After 200 Wave Impacts

Figure 4: Sample Simulation Process

3.7 Calculate & Simulate Results

Briefly review the equation derived in section 3.1

$$tan\theta_c = \mu + \frac{\sqrt{2}\pi\mu\Gamma}{R\rho gD} sec[tan^{-1}(\mu)]$$

To solve for the best shape of the slope, the necessary information about sand is listed below

Table 5: States of Sand

Physical Quantities	Values	Units
Γ	72.8	mN/m
μ	0.55 0.60	-
ρ	1631	kg/m^3
R	0.05 2	mm
g	9.81	m/s^2

It is not sensible to embark on a trip to the beach on a stormy day, so we assume that the near-shore wave is gentle, with its height not exceeding 20cm, or $D \in [0, 20](unit : cm)$. We adopt $\Gamma = 72.8mN/m$, $\mu = 0.55$, $\rho = 1631kg/m^3$, $R = 2mm$ and $g = 9.81m/s^2$.

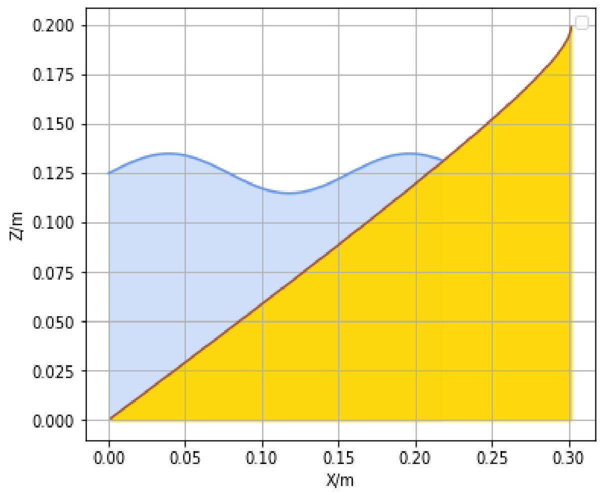


Figure 5: Slope Shape

Table 6: States of Sand

Height(m)	Slope
0.02	0.58
0.04	0.58
0.06	0.59
0.08	0.60
0.10	0.60
0.12	0.61
0.14	0.63
0.16	0.66
0.18	0.71
0.20	0.88

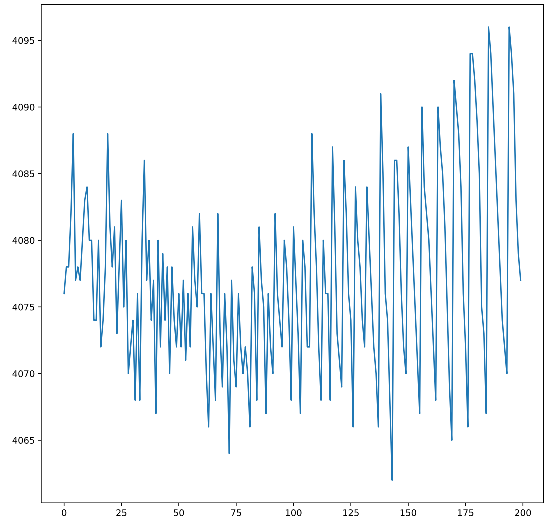


Figure 6: Iterations Until Collapsing Over 200/2000 Samples of Permutaion

3.8 Permutation

We try to permute the possible arrangements of fixed-sand-amount sand castle because of the discrete property of a **cellular automata**. To level down the computation required, we think of these three strategy:

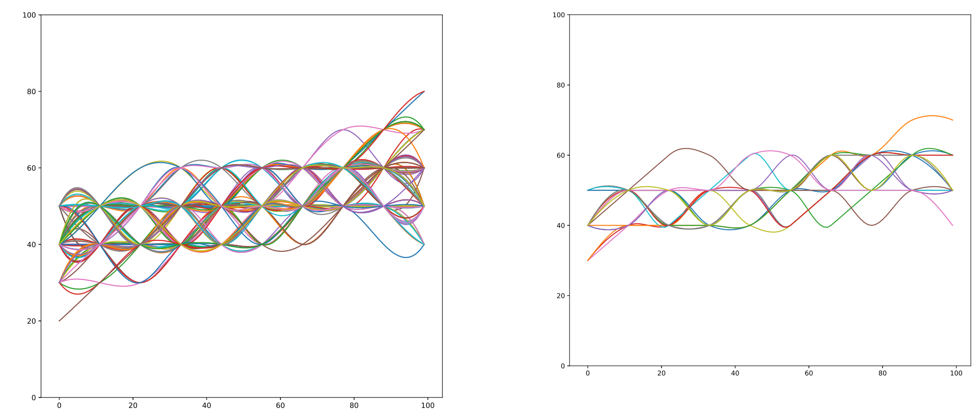
1) Permute only the possible arrangements on a small simulation field. Then, gradually increase the resolution without having to exponentially raise the number of permutations to compute on.

2) Use interpolation to convert permutations from low resolutions to a finer edge definition. Then simulate on the upgraded arrangements.

3) Make more assumptions. For example, if we restrict our sand castle's front edge to be second-order derivable, and estimate curves with a distorted circle, we can significantly reduce the permutations to test on.

Let denote the width of the simulation field as W and depth as D. If all possible permutations should be considered, the total number of them would be D^W, which is totally unacceptable and unimaginable. Thus we'd like to restrict the 1st order derivation to be in a specific bound: $[-\mathit{\Delta}, \mathit{\Delta}]$, which will lead to a total permutation of $(2 * \mathit{\Delta} + 1)^W$ ruffly. Let's say $\mathit{\Delta} = 1$, and $W = 10$. So the total number of permutation would be ruffly $196,830$, which is temporarily acceptable for a modern computer. However, during the process of running simulation, we've discovered that **ten** is too small a number to represent the real world sand permutation, as shown in Figure 6. Unfortunately, the barrier between 3^10 and 5^20 (if we double the amount of $\mathit{\Delta}$ and W) is able to be compared with $20! - 10!$.

So we turn our attention to interpolation, hoping to generate a reasonable simulation result by adding enough resolution. However, as you can see in Figure **??**, both **Cubic Spline** and **Lagrange Interpolation** leads to unacceptable results.



(a) 100/400 Best Case: Cubic Spline

(b) 10/100 Best Case: Lagrange Interpolation

Figure 7: Interpolation

Eventually, by adding the assumption:

$$Allsandcastle'sfrontedgearesecond-orderderivable, and,$$
$$Arbitrarycurvescanbeestimatedbydistortedhalfcircle, and,$$
$$Weexpectonlyonemaxvalueinthecurveofsandcastleedge, and,$$
$$Curvesaretheonlyformofedgeconsideredsincefluidmaysmoothenedge$$

We can compute out the best circle-like curve from the permutations shown in Figure 8 (each line represents an edge set up).

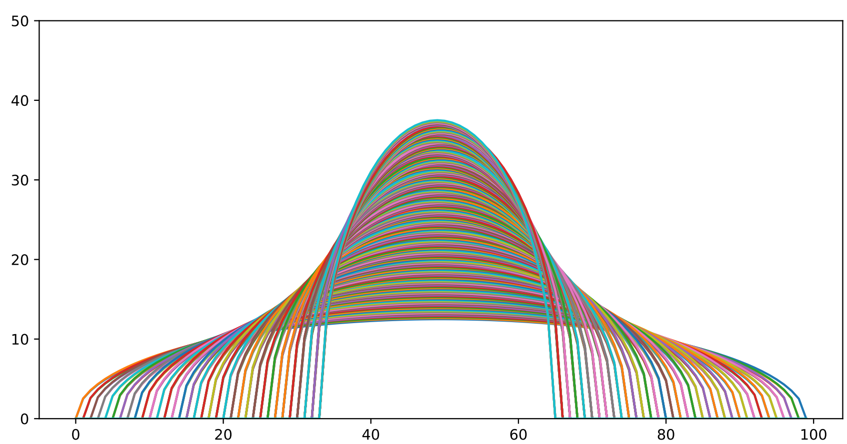


Figure 8: Distorted Circles

Then result of those sets of simulations and estimations is shown in Figure 9 and Figure 10.

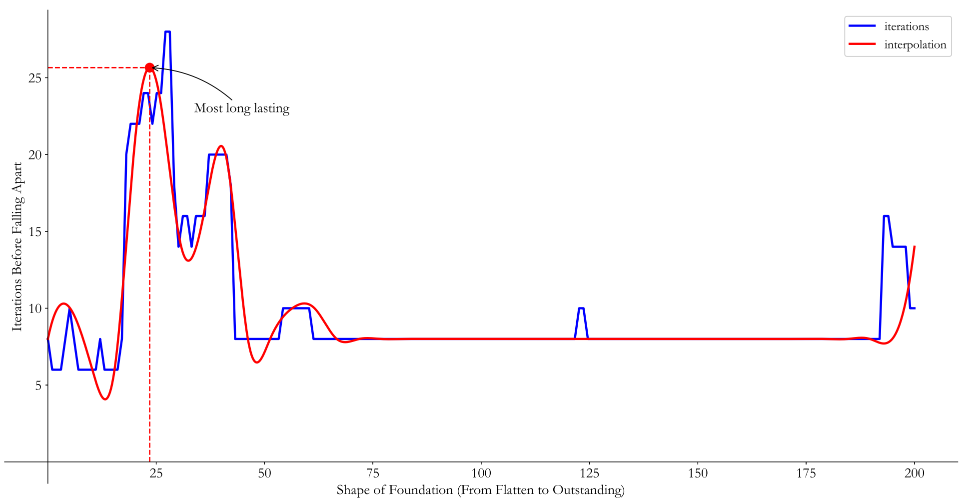


Figure 9: Iterations Until Melt Down

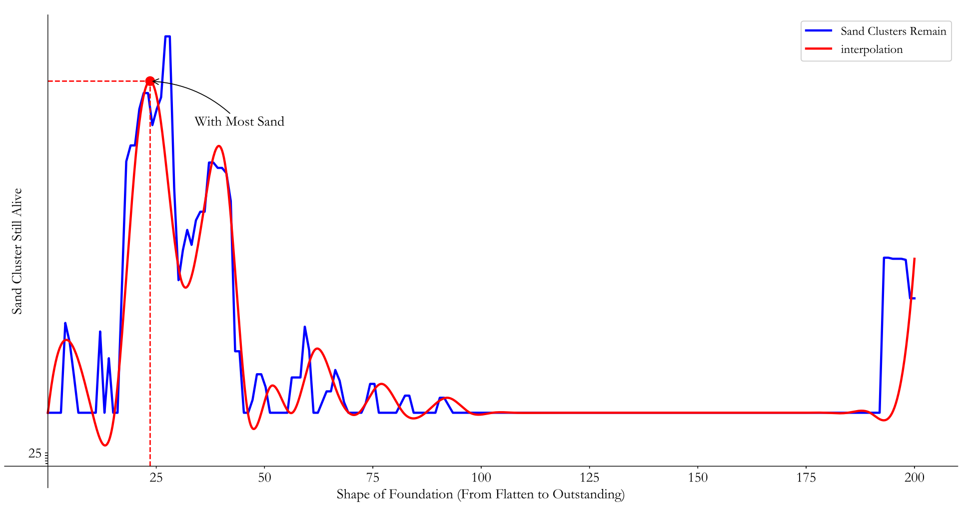


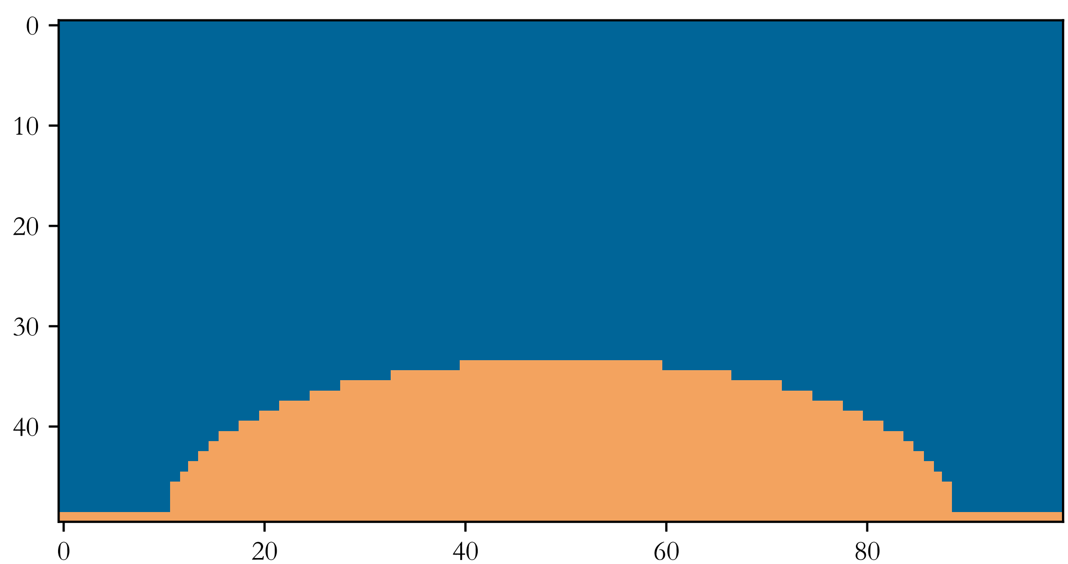
Figure 10: Remaining Sands After Fixed Amount of Iterations

The final conclusive result would be shown in Figure 11 and Table 7.

Table 7: Best Top View (width=100)

0.	0.	0.	0.	0.
0.	0.	0.	0.	0.
0.	3.46899837	4.87435499	5.93094938	6.80326553
7.55540338	8.22041701	8.81803996	9.36118789	9.85886839
10.31766302	10.74255518	11.13742567	11.50536599	11.84888516
12.17005135	12.4705917	12.75196464	13.01541332	13.262006
13.49266687	13.7082002	13.90930923	14.09661133	14.27065016
14.43190557	14.58080168	14.71771362	14.84297302	14.95687263
15.05967022	15.15159174	15.2328341	15.30356738	15.36393672
15.41406389	15.45404855	15.4839693	15.50388438	15.51383233
15.51383233	15.50388438	15.4839693	15.45404855	15.41406389
15.36393672	15.30356738	15.2328341	15.15159174	15.05967022
14.95687263	14.84297302	14.71771362	14.58080168	14.43190557
14.27065016	14.09661133	13.90930923	13.7082002	13.49266687
13.262006	13.01541332	12.75196464	12.4705917	12.17005135
11.84888516	11.50536599	11.13742567	10.74255518	10.31766302
9.85886839	9.36118789	8.81803996	8.22041701	7.55540338
6.80326553	5.93094938	4.87435499	3.46899837	0.
0.	0.	0.	0.	0.
0.	0.	0.	0.	0.

Figure 11: Best Top View



4 Modeling Under Rain

4.1 Model Adaptations

Rain differs from waves and sands in the following aspects: its continuity, and its quantity. So we modify our Cellular Automata model to adapt to the new conditions.

Adaptations

1) In our previous model, wave is modeled as discrete motions.

2) Rainwater can wet our model in a relatively short time.

3)

4.2 Shape Alteration

As a result, our model is not the most preferable option under the rain. Shape alterations have to be made if we want our sand castle foundation to last longer.

5 Determine the Best Sand-to-water Proportion

5.1 Assumptions & Nomenclature

ADD A CURVE FIGURE HERE, DRAWING!

As we mentioned above, the adhesive stress across the plane (s_A) is determined by water content. Consequently, we will find out how the cohesion in wet granular materials varies with different amount of liquid based on liquid bridge model to find an optimal sand-to-water mixture proportion.

There are five differing degrees of wetting in granular matter. As dry grains and slurry have no cohesion between particles, we just consider three other conditions. According to [2], we can see that the slope for low and large liquid content is much larger than that for intermediate liquid content. That is: the suction changes sharply as these three states transform each other. Consequently,we should keep the proportion of water to sand in steady for a long time, which means try to avoid transformation between three states.

We initially wished to create a three dimensional matrix to represent the situation. We still consider that the sandcastle C consists of cells mixed with sand and water. Then It's easy to Calculate each cell's sand-to-water mixture proportion from the amount of sand and water. ADD TWO CELLULAR FIGURES HERE, DRAWING!

5.2 Assumptions

1) waves and tides erode the sandcastle in a very short time, which means that the we can neglect Osmosis occurred during erosion.

2) After eroded, the liquid left on sandcastle has ample time to penetrate.

3) The evaporation of liquid is neglected.

5.3 Review on Nomenclature

Table 8: Notation Table

symbol	meaning
$c_{ijk}(t)$	the cell mixed with sand and water to form the sandcastle in time t
$s_{ijk}(t)$	the amount of sand in the cell in time t
$w_{ijk}(t)$	the amount of water in c_{ijk} in time t
$p_{ijk}(t)$	the proportion of sand-to-water mixture c_{ijk} in time t
$S(t)$	the amount of sand in the sandcastle int time t
$W(t)$	the amount of water in the sandcastle in time t
$P(t)$	the average proportion of sand-to-water mixture c_{ijk} in time t
$F(t)$	the erosion-Osmosis common effect of the sea water in time t

5.4 Model

We define the expression when we just finish build the sandcastle.

$$P(t_0) = \frac{S(t_0)}{W(t_0)} = \frac{\sum_{i,j,k} s_{ijk}(t_0)}{\sum_{i,j,k} w_{ijk}(t_0)} = p_{ijk}(t_0)$$

To evaluate the existence of the sandcastle, we define that a cell is completely eroded by tides and waves as follow:

$$p_{ijk}(t) = \frac{s_{ijk}(t)}{w_{ijk}(t)} \leq p_{erode}$$

And we define that the sandcastle is completely eroded as:

$$P(t) = \frac{\sum_{i,j,k} s_{ijk}(t_0)}{\sum_{i,j,k} w_{ijk}(t_0)} \leq P_{erode}$$

where P_{erode} refers to proportion that Funicular transforms to capillary state. Then we define the flow equation during erosion as follow:

$$E = \begin{cases} F & \text{the cell is being eroded by sea} \\ 0 & \text{else} \end{cases}$$

As erosion occurs for a split second, we could just neglect the permeation to simplify our analysis. To obtain a probability model to evaluate the sand taken by waves, we need to acquire further understanding about the flow liquid. Here we use Navier-Stokes equations to describe the motion of viscous fluid substances.

$$x\ \rho(\frac{\partial \upsilon}{\partial t}+\upsilon\cdot\nabla\upsilon) = -\nabla p+\nabla\cdot(\mu(\nabla\upsilon+(\nabla\upsilon)^T)-\frac{2}{3}\mu(\nabla\upsilon)I)+F\frac{\partial\rho}{\partial t}+\rho\nabla\cdot\vec{v} = 0(continuity\ equations)$$

After eroded we should find the cell completely eroded by tides and waves. Then by reference to Darcy's law and Huygens–Fresnel principle, We consider water permeating the sandcastle as follow:

$$\upsilon = \frac{kA(p_b - p_a)}{\mu L} = \frac{kL(\Delta P)}{\mu}$$

where k is the hydraulic permeability of sea water, L is the length of one side of the cell, μ is the viscosity of sea water, Δ is the pressure differential between the two sides of the cell.As we assume that the sea water left on the surface of the sandcastle can totally penetrate. Then we could get for each cycle:

$$L(k,t,\mu,\rho,\upsilon,p) = \sum_{c_{ijk}\in C}\sum_{i-1}^{i+1}\sum_{j-1}^{j+1}\sum_{k-1}^{k+1} p_{mnp}\cdot E(m,n,p\neq i,j,k)(P(t)<P_{limit})$$
$$(\upsilon\cdot\nabla\upsilon)+(\mu\cdot\nabla\mu)+(\rho\cdot\nabla\rho) = Constant$$

Then We need to find out the longest time when the above constraints are met and the sandcastle exists.

6 Other Tricks to Make Our Sand Castle Last Longer

6.1 Better conditions to survive

As we acknowledge that the greatest threats to our sandcastle is the erosion of tides and wives. Consequently, a relatively safe place will be better if we want to protect it from erosion for a longer time. That means we can do something as follow:

Build a Moat Building a moat can help our sand castle foundation to survive the rising tide.

Further From the Shore Just stay in the "safety zone" and don't build your sandcastle too close to the shore will obriously reduce the damage caused by tides and waves.

Less Salty Water Choosing a beach near a less salty ocean may contribute to building a more concrete sand castle. This is because fresh water has a higher value of surface tension which appears in the numerator of the term $\frac{\sqrt{2}\pi\mu\Gamma}{R\rho g D}$. Thus Red Sea is not a perfect spot, but the Baltic Sea(poor English. better expression needed.)(footnote needed)

6.2 Sharpen Your Design

Make It Compact In section 3.1, we assumed that sand particles are closely packed. In fact, there are some degree of internal erosion in the sandcastle along with the hydraulic pressures increase. The final value of the hydraulic conductivity also depends on the rate of erosion of the liner material.
According to study [4], The erodibility of a soil can be described in term of the rate of erosion. They find that Correlation between erodibility and Hydraulic Conductivity of Sand-Bentonite Mixtures can be expressed in the following form:

$$k = a(\varepsilon_e)^b$$

where a,b depends on the percentage of bentonite in the mixture (η).These studys indicate that the rate of erosion depends mainly on the porosity of the mixtures.Consequently,you had better make your works more compact to defend it from internal erosion.

Finer Sand Finer sand is a bonus because by constructing our sand castle foundation using finer sand, the foundation has a greater critical angle. That is because the radius of particles R which is a part of the denominator decreases and (enlarge?not the proper word) the value of $tan\theta_c$.

Special Material

Glue Sand sculptures seen in exhibitions or competitions have their surfaces covered with special glue. The glue is mainly used to protect the sand sculpture from drying out or crumbling if the waves do not wash it away first, but it can also keep water out. In our model where the greatest enemy of the foundation is not collapses but erosion, such glue will help maintain the shape of the foundation.

Mixture Concrete, a material that everyone is familiar with, has sand as its ingredient.

7 Sensitivity Analysis

7.1 Model Under Waves and Tides

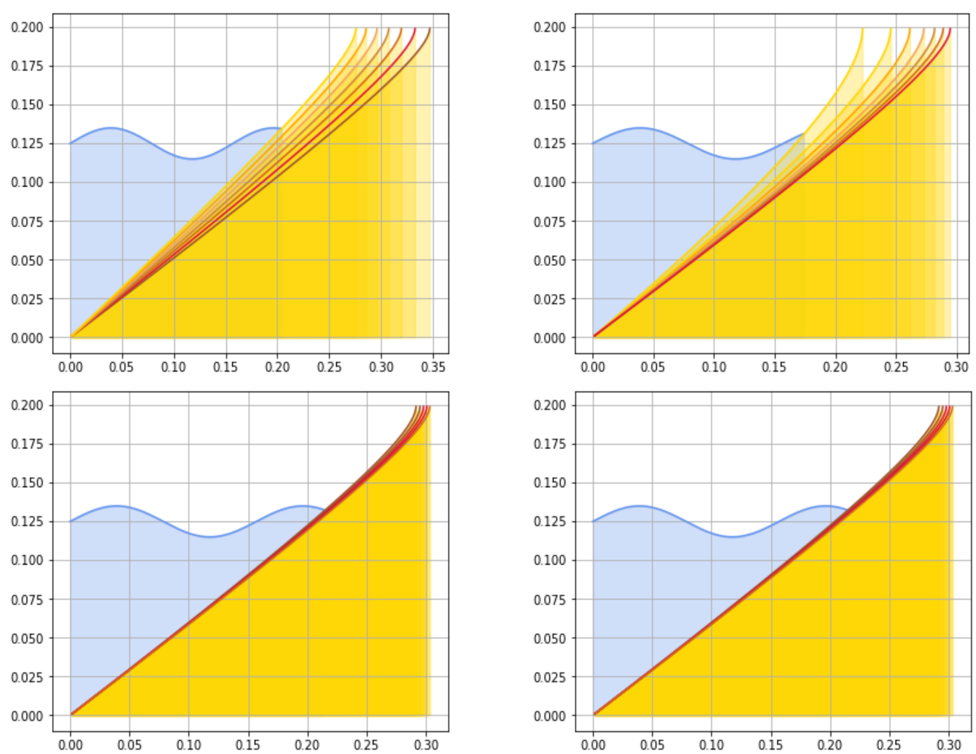
Side View Shape There are multiple factors that may affect our side view shape model. We consider four of them: the internal friction coefficient μ, radius of the sand particles R, density of sand ρ and total height of the sand castle foundation D.

Friction coefficient of sand varies by the amount of liquid in the system, the size of sand grains and other possible factors. Typically, this value ranges from 0.55-0.60. In figure[x.a], we test the effect of different friction coefficient ranging from 0.48 to 0.60. The curves with lighter colors, as illustrated in the figure, represent higher coefficients while those with darker colors represent lower coefficients.

Figure[x.b] shows how the side view shape is influenced by radius of the sand particles, which varies between 0.5 to 2.0 milimeters. Curves with lighter colors represent small particle radiuses and those with darker colors represent greater radiuses.

The effect of density is illstrated in Figure[x,c], which is not obvious compared with the former two factors.

Table 9



Top View Shape DenDen dl has done a lot of parameter adjusting jobs...those are exactly what we need here.

7.2 Model Under Rain

Not now. Wait for tomorrow or something.

8 Strengths and Weaknesses

8.1 Strengths

Just like any other model, the one presented above has its Strengths and Weaknesses. Some major points are listed below.

1) **Flexibiliry and extendibility**
   The use of Cellular Automata enables our model to predict erosion of any possible shape. For those who are not satisfied with a dull, oval-like shape, our model can assist them in finding more interesting shapes with relatively small losses.

2) **Taking Future into Consideration**
   Cellular Automata can simulate the whole life-cycle of the sand castle foundation. Thus we are judging a shape by not only its current performance, but also how it performs after its shape has been changed.

3) **Taking into Consideration the Height of Waves**
   Due to tidal effects, the height of the waves is not a constant. From the analysis in section 3.1, the shape identified can resist waves from t

8.2 Weaknesses

1) **Not accurate enough**
   Our model in section 3.1 is actually an approximation of reality, as in capillary state, the amount of water is very close to saturation, which means liquid bridges only play a minor role in stabilizing the sand castle foundation. Even though in Funicular state, where the cohesive stress s_A increases monotonically with water content for most of the time, the two states differ in properties and will bring loss to our model.

2) **Simplistic Assumptions**
   To acquire computability, we simplify the effect of waves to be wetting and hitting the surface. Nevertheless, the flow direction is also of vital significance, about which we fail to fully consider.

9 Future Developments

We can

List of Figures

List of Tables

List of Listings

References

[1] M. Pakpour, M. Habibi, P. Møller, and D. Bonn, "How to construct the perfect sandcastle," *Scientific reports*, vol. 2, no. 1, pp. 1–3, 2012.

[2] N. Mitarai and F. Nori, "Wet granular materials," *Advances in Physics*, vol. 55, no. 1-2, pp. 1–45, 2006.

[3] A. Kudrolli, "Sticky sand," *Nature materials*, vol. 7, no. 3, pp. 174–175, 2008.

[4] J.-H. Yoo, H.-S. Lee, and M. A. Ismail, "An analytical study on the water penetration and diffusion into concrete under water pressure," *Construction and Building Materials*, vol. 25, no. 1, pp. 99–108, 2011.

[5] *Huygens–fresnel principle - wikipedia.*