3D rock-fall simulation using CELL-DEVS model

Pei Jin Yang An

Department of Electrical and Computer Engineering
University of Ottawa
550 Cumberland Street
Ottawa, ON. K1N-6N5 Canada
{pjin049, yan073}@uottawa.ca

ABSTRACT: When a theory is proposed for discrete-event system, it means that simulation of a complicated system can be resolved by dividing the whole system into hierarchical and modular components. In this way, the process of simulation can be improved significantly due to its advantages in many aspects, such as the reuse of model, cost of maintenance and test times. As the development of Cell-DEVS, it has been used in many practical systems, such as computer networks, emergency evacuation, traffic controllers, and flexible manufacturing plants. In our simulation, we will use Cell-DEVS model to simulate the commonplace natural phenomenon- rock fall in 3D version. Each cell presents a stone on the surface of the mountain which is divided into 3 zones- slopes with different degrees (75, 45 and 15). Each cell has different rules and neighbors in three zones. Also, we compared and analyzed how the simulation behaves when different values of state variables are assigned to each cell.

1. Introduction

Simulation is model execution which is presented by computer coding program generally describes the real system. As using real system to simulate is too expensive, simulation by a computer can really save time and cost. In recent years, a number of modeling techniques have been introduced in order to improve the definition and analysis of complex dynamic systems. The development of simulation tools has often been closely related to the execution of these models [1].

Cell-DEVS, as one of discrete-time simulation models, is very commonplace now, which can be seen from many fields, such as, traffic control, passengers boarding and intellectual games and is also used to simulate some natural phenomenon like fire spread, mining, etc.

Rock fall is a quite normal phenomenon in the natural activities, which poses a threat to the mountain area all over the world. It is a hard work to predict when it will happen because different areas or regions have various rock properties. Once it happens, it has a huge damage to the humanity, animals and the nature.
A rock fall starts from a loose rock falling along vertical and sub-vertical paths together. Apparently, as for the vertical path, the velocity can reach a very high speed by accumulating stones. The much heavier rock is, the faster speed has. In terms of sub-vertical rock fall, it can only get a relatively small horizontal speed brought by the collision of near stones at first, while as they are rolling all the time and they can be fast as well finally before they reach the bottom of mountain.

In our simulation, we will use Cell-DEVS model to simulate the process of rock-fall in 3D version. And we will analyze different results and compare different zones of the mountain with different slope degrees, ranging from 15, 45 to 75. In addition, in order to simulate the process of rock-fall approaching to the reality, we also add many different state variables- motionless, toppling, rolling, bouncing and falling.

2. Background

2.1 DEVS model

Discrete simulation is the modeling of a system as it evolves over time by a representation in which the state variables change only at a countable number of points in time. At these points an event occurs, where an event is taken to be an instantaneous occurrence that may change the state of a system [2]. As a large number of models are not static and they will change with the time moving, lasting from this time point to that one, a discrete time simulation models are required.

The basic DEVS model can be regarded as a integration of sub-models, which can represent a whole system. Each model can be executed and it has input, output, state, time and function. The DEVS model can be represented as:

$$M = < I, X, S, Y, \delta int, \delta ext, \lambda, D >$$

Here, **I** is the model's interface, **X** is the input events set, **S** is the state set, and **Y** is the output events set. There are also several functions: **δint** manages internal transitions, **δext** external transitions, **λ** the outputs, and **D** the elapsed time [3].

2.2 Cell-DEVS model

Apparently, timed Cell-DEVS model is more complicated than the DEVS model. Its atomic model can be presented as the following:

$$TDC=< I, X, Y, \theta, N, delay, d, \delta int, \delta ext, \tau , \lambda, D>$$

I is the model's interface, **X** is the set of input events, and **Y** is the set of output events. **D** is an index of components, and for each **i** ∈ **D**, **Mi** is a basic DEVS model (that is, an atomic or coupled model). **Ii** is the set of influences of model **i**. For each **j** ∈ **Ii**, **Zij** is the **i** to **j** translation function. Finally, select is a tie-breaking function defining which model executes if more than one is activated simultaneously [3].

The time advancement is also necessary in our simulation model, because the process of collision of different sizes and shapes of stones is different. In addition, the stones in different slopes of the mountain are various as well. So we need use corresponding delays to represent state changes of different stones.

2.3 Benefits of RISE in our model

All parameters are different and difficult to quantify and calculate in real world. So a lot of simulators choose to use random components to test the rock fall situation due to local conditions. After reading these papers we decide to use different random probabilities to represent the combination of different parameters of nature geographic information. We know that one fall rock can cause different affects to different dimensions, which we use different probabilities to represent in this project. Here are some explains about the idea which is the foundation of this simulation.

Thanks to the benefits of RISE version of the software, we can add neighbor ports to each cell and set more state variables like motionless, toppling, rolling, bouncing and falling rather than only two choices in the old version of CD++ version.

So here, after we learned from the paper [4], we know that there are several parameters which may affect the result of the simulation. For instance, it varies dramatically and is unpredictable once we take initial velocity and kinetic energy, shapes and sizes of stones and friction between stones and the surface of the mountain into consideration.

2.3.1 State variables

We used 3D dimensional geometry to depict the scenario of a falling rock. We use 20*20*20 cubes to implement the environment.

We initialize the whole cubes with the value 0, which represents the stones on the slope of the mountain are stable. Then, the value of cube will begin to change to different values according to change of its neighbors, which indicts the motionless stone on this slope begin to change to other states- toppling, rolling, bouncing and free-falling. Obviously, it has a great improvement than the old CD++ version, which can only have one state variable, really limiting the scope of application.

2.3.2 Add neighbor ports

It is easy to understand that different possible results will generate when a stone is collided with a falling stone. Since the probability that a stone will keep motionless or change to topple, roll, bounce or fall is difficult to predict in the real world, we simply it by calculating 5 neighbor ports of each cell to determine which state the stone will change to.

Finally, the stone should have different velocities as it may be affected by different stones from different directions, so different time delays are used to describe this case. And these delays can be understood easily. For instance, the shortest delay is the case that the stone is collided by the upper falling stone.

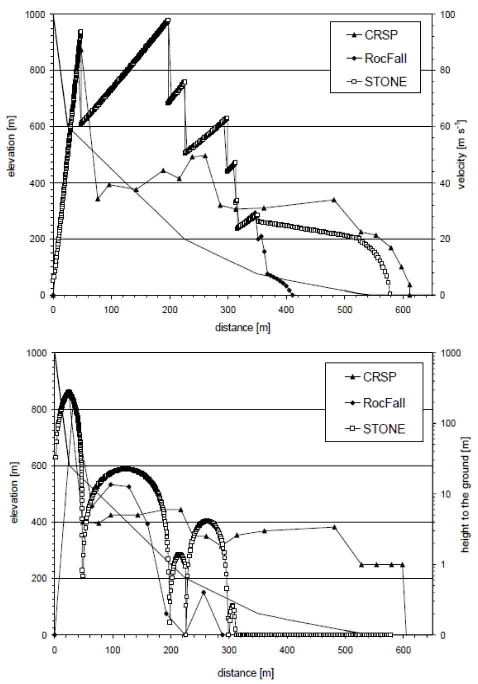

Figure 1. Effects of different heights of stones on their speeds when they start to fall

Figure 1 mainly shows that on different heights, the falling stones have different speeds. The much higher the height is, the faster the speed is. Then, we change this height to relative height and distance. For instance, because the relative height between the upper stone with the lower one neighbour is the biggest, so we can assume its speed is the fastest and the momentum is the biggest. So, as for this direction, it has the shortest delay time and in the same way, it will affect next lower stone with largest probability.

2.3.3 Different zones

In order to make the simulation approaching to the real world, we divide the whole surface of the mountain into 4 different zones- 15, 45 and 75 slope degree of the mountain.

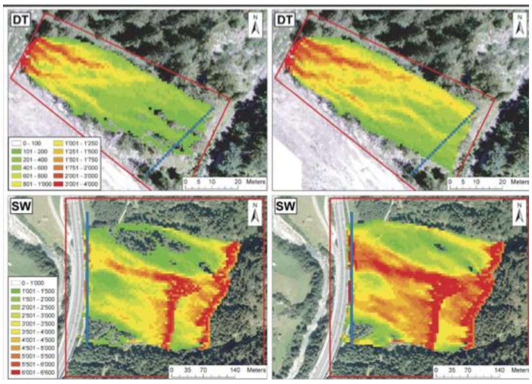

Figure 2. Rugged topography of the real mountain

As can be seen in the figure 2, the real mountain has different slopes. However, if it is not inerratic in one zone, it will be difficult to define cell's neighbors and cannot guarantee each cell has the same behavior. Consequently, the whole mountain will probably collapse. So in each zone, we just consider the stones only with changes of position in the axis x.

a	**a**	**a**	**a**	**a**	**a**	**a**	**a**
			1	2			

Figure3. Plane x

In this case, if we see the plane x regardless of which zone (15, 45 and 75) it belongs to, we will not take stones 1 and 2 into consideration in order to define the same behavior of each cell a.

The zone which has 0 degree slope means it is the bottom of the mountain. The normal result of rock fall in nature is that a huge amount of stones is accumulated on the bottom. In terms of the process of simulation, the value of cell in the plane will have a change, it is really easy to implement, but we do no choose to do it finally because it will add more cells which will take a long time to simulate. Also, because our model is divided into zones, so we should give at least one initial cell in each zone, but it will not make sense in that zone.

3. Model definition

3.1 Take 45 degree slope (zone 2) as an example

Cell-DEVS formal specification

M = < Xlist, Ylist, I, X, Y, S, θ, N, d, τ, δint, δext, λ, ta >

Xlist = { Φ}
Ylist = { Φ}
I = < 6, 0, Φ, Φ>
X = Φ
Y = Φ

S = $\{s | s \in (0, 6)\}$

θ = {(s, phase, σqueue, σ), s $\in$ S, phase $\in (0, R)\}$
N = {(-1,0,0), (0,0,0) (1,0,0), (1,-1,-1), (0,-1,-1), (-1,-1,-1) }

d = 100 ms
ta (passive) = INFINITY
ta (active) = d

Figure 4. Formal specification of simple rock-fall model

Figure 4 shows the simple formal specification of rock-fall model. The simple model means the whole slope of the mountain is 45 degrees with the horizontal line. In this simple model, we just need to define the 5 neighbors of each cell ((-1,-1,-1), (0, -1, -1), (1,-1,-1), (-1,0,0), (0,0,0) (1,0,0)). Because we only focus on the part of slope of the mountain, so it should not be wrapped.

In our model, assuming the point (0, 0, 0) is the stone that will be checked whether it will begin to move, which state it will change to-motionless or start to roll according to the values of its 5 neighbors which means 5 stones adjacent to it. And the detail of rules of each cell will be explained in the next part of model definition.

As we can see from the YoZ platform, the 5 neighbors ((-1,-1,-1), (0, -1, -1), (1,-1,-1), (-1,0,0), (0,0,0) (1,0,0)) are listed in the Figure 5.

(-1,-1,1)	**(-1,0,0)**	**(-1,1,-1)**
(0,-1,1)	**(0,0,0)**	**(0,1,-1)**
(1,-1,1)	**(1,0,0)**	**(1,1,-1)**

Figure 5. From the YoZ dimension

One thing should be noticed is that each row of the figure 5 does not belong to the same plane. For instance, the first row (-1, -1, 1), (-1, 0, 0) and (-1, 1, -1) is in the upper plane, the second row (0, -1, 1), (0, 0, 0) and (0, 1, -1) is the middle one and the third row (1, -1, 1), (1, 0, 0) and (1, 1, -1) is the lower one. So there are 5 neighbors for each cell in different planes, but the physical constant that is multiplied with received energy they transfer to the stone (0, 0, 0) is ranging from 0.23, 0.38 to 0.50 according to the reasonable circumstance. In addition, all of stones' initial values are set to 0. If the affected stone falls, the value will change from 0 to 1 (falling). Then the falling stone will bounce or roll depending on the shape of stone.

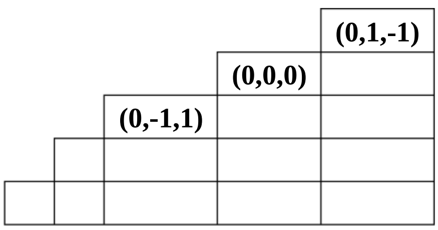

Figure 6. From the (XoY) dimension

In the figure 6, we see the whole model from the XoY dimension rather than the YoZ, as the result, 3 cells are in the figure, in which only 1 cells (0, -1, -1) that can affect the cell (0, 0, 0) can be seen. It is easy to understand that the stone (the cell (0, 1, -1)) starts to fall can affect the stone (the cell (0, 0, 0)). Although it does make sense because, in the real world, the lower stone supports the upper stone, which is regarded as the foundation, we do not take it into consideration, that is, we do not take them as (0,0,0) neighbors.

3.2 Rules of model

The rule of this simulation model is based on the basic physical principle that a motionless stone starts to fall from the static state. Specifically, we assume the stone is triggered by the absorption of the kinetic energy from neighbor stones. Then, the falling stone will be determined to keep to static state or change to topple, bounce or roll according to the numerical difference value between the kineticEnergyVal and frictionVal, which are state variables. Another state variable should be noticed is stateVal, which represents the state of a cell. The details of behavior of a cell, that is, the change of states, will be explained in 3.2.1, 3.2.2, 3.2.3 and 3.2.4.

3.2.1 The basic model definition

```
[FallRock]
type : cell
dim : (8,8,8)
delay : transport
defaultDelayTime : 100
border : nowrapped
neighbors :        FallRock(1,-1,-1)
FallRock(0,-1,-1) FallRock(-1,-1,-1)
neighbors :        FallRock(1,0,0)
FallRock(0,0,0)    FallRock(-1,0,0)
neighbors :        FallRock(1,1,1)
FallRock(0,1,1)    FallRock(-1,1,1)
```

Figure 7. (a)FallRock model definition

```
initialValue : 0
initialCellsValue : init.val
StateVariables : stateVal kineticEnergyVal
frictionVal shapeVal countVal
stateValues : 0 0 0 0 0
InitialVariablesValue : initial.stvalues
localTransition : FallRockBehavior
```

Figure 8. (b) FallRock model definition

As can be seen from the figure 8, the initial value of each cell is 0 and some cells should not be zero, while represents the stones that trigger the rock fall. However, we do not assign them special values and the default value of all cells is 0. Instead, we use state variable (stateVal) to determine the change of cell's state. That value will be given in the initial.stvalues file which is used to assign initial values of state variables of all cells.

3.2.2 Rules of different cell states in the 45 degree slope (Zone 2)

```
%Motionless
rule :
{
       ( 0 - $frictionVal )
}
{
       $stateVal := 0 ;
       $kineticEnergyVal := 0 ;
}
100
{
            (( 0.98689 * $kineticEnergyVal +
              +0.23*(1,-1,-1)~neighborChange
              +0.50 * (0,-1,-1)~neighborChange
              + 0.23 * (-1,-1,-1)~neighborChange
              + 0.38 * (1,0,0)~neighborChange
              + 0.38 * (-1,0,0)~neighborChange
              ) < $frictionVal
             )    AND
             ($stateVal = 0 )
}
```

Figure 9. The rule of motionless

As for the second zone of the mountain, the slope degree is 45 with the horizontal line, under the first zone (75 degree) and above the last zone (15 degree). So the neighbors of (0,0,0) are (1,-1,-1), (0,-1,-1), (-1,-1,-1), (1,0,0) and (-1,0,0). The condition of the cell in this zone is that the initial stateVal is 0 which means it stay motionless. If it cannot obtain enough kinetic energy from its neighbors smaller than the resistance, it will keep motionless and the kinetic energy of itself is 0. And we output negative ten to its neighbors, differentiating with the state-toppling. Also, it will make sense when computing its neighbor cell because it means that the neighbor cell will get a negative value from it, representing the neighbor cell will need more kinetic energy to make the two motionless stones to begin to move.

```
%Toppling
rule :
{ 0 }
{ $stateVal := 1 ; $kineticEnergyVal := 0 ;
}
100
{ (( 0.98689 * $kineticEnergyVal
        +0.23*(1,-1,-1)~neighborChange
        +0.50 * (0,-1,-1)~neighborChange
        + 0.23 * (-1,-1,-1)~neighborChange
        + 0.38 * (1,0,0)~neighborChange
        + 0.38 * (-1,0,0)~neighborChange
          ) > $frictionVal
         ) AND
         (( 0.98689 * $kineticEnergyVal
        +0.23*(1,-1,-1)~neighborChange
        +0.50 * (0,-1,-1)~neighborChange
        + 0.23 * (-1,-1,-1)~neighborChange
        + 0.38 * (1,0,0)~neighborChange
        + 0.38 * (-1,0,0)~neighborChange
          ) < ($frictionVal+20)
         ) AND
         ($stateVal = 0 OR $stateVal = 1 )
}
```

Figure 10. The rule of toppling

The prerequisite of this rule of the cell in this zone is that the initial stateVal is 0 (means it is motionless) or 1 (means it is toppling). In short, the stone does not leave its place. And then, if it obtains some kinetic energy from its neighbors larger than the resistance but smaller than the resistance plus 20, the stateVal will change from 0 to 1, which means the state of the stone changes from motionless to toppling or the sateVal will keep toppling and in both cases, the kinetic energy of itself is 0. And we output zero to its neighbors, differentiating with the state-motionless.

```
100
{
         (( 0.98689 * $kineticEnergyVal
        +0.23*(1,-1,-1)~neighborChange
        +0.50 * (0,-1,-1)~neighborChange
        + 0.23 * (-1,-1,-1)~neighborChange
        + 0.38 * (1,0,0)~neighborChange
        + 0.38 * (-1,0,0)~neighborChange
        ) > ($frictionVal + 20)
         ) AND
         (
            $stateVal = 0 OR $stateVal = 1
         )
}
```

Figure 11. The condition of rule of falling

The prerequisite of this rule of the cell in this zone is that the initial stateVal is 0 (means it is motionless) or 1 (means it is toppling). In short, the stone does not leave its place. And at the same time, if it obtains enough kinetic energy from its neighbors larger than the resistance plus 20, it will change state.

```
%falling
rule :
{
        (( 0.98689 * $kineticEnergyVal
        +0.23*(1,-1,-1)~neighborChange
        +0.50 * (0,-1,-1)~neighborChange
        + 0.23 * (-1,-1,-1)~neighborChange
        + 0.38 * (1,0,0)~neighborChange
        + 0.38 * (-1,0,0)~neighborChange
          ) - $frictionVal)
}
{
        $stateVal := 2 ;
        $kineticEnergyVal :=
        ((0.98689*$kineticEnergyVal
        +0.23*(1,-1,-1)~neighborChange
        + 0.50 * (0,-1,-1)~neighborChange
        + 023 * (-1,-1,-1)~neighborChange
        + 0.38 * (1,0,0)~neighborChange
        + 0.38 * (-1,0,0)~neighborChange
        ) - $frictionVal) ;
}
```

Figure 12. The rule of falling

After the time delay- 100, the stateVal will change from 0 to 2 and the energy of this cell will change to the obtained energy distracted by its resistance and store them into its state variables- stateVal and kineticEnergyVal, respectively. Then, the cell outputs the remaining kinetic energy to its neighbors.

In the real world, this rule presents the moment that the stone is collided with the rolling stone and become separated from the adjacent stones. Actually, it is the state before the state- rolling. And at this moment, it has enough kinetic energy.

```
%bouncing
rule :
{ 0 }
{
    $stateVal := 4 ;
    $kineticEnergyVal :=
    (0.98689*$kineticEnergyVal+0.50) ;
}
100
{
     ( $stateVal = 2 OR $stateVal = 4 )
AND
             ( $shapeVal = 1 )
}
```

Figure 13. The rule of bouncing

This rule is used to represent the bouncing stone which has an irregular shape. When it is collided with other stones, it could not move as the stone with regular ones. So the track of moving is bouncing rather than rolling in the regular path defined initially.

```
%rolling
rule :
{
        (0.98689 * $kineticEnergyVal )
}
{
        $stateVal := 3 ;
        $kineticEnergyVal :=
        (0.98689 * $kineticEnergyVal ) ;
}
100
{
    ( $stateVal = 2 OR $stateVal = 3 ) AND
        ( $shapeVal = 0 )
}
```

Figure 14. The rule of rolling

This rule is quite different from the above three rules. In fact, it is the rule that will trigger the whole simulation. Its state variable- stateVal is 2 or 3 which means falling or already rolling. Also its shape should be 0 which means the shape is regular. That is totally opposite to the rolling state. After 100 time delay, it will outputs its kinetic energy. And it transfers its stateVal to 3 and stores its energy in the state variable.

In our simulation, we will assign the value- 2 to one of the cell which means it is the first moving stone. And then it will output its kinetic energy. At next time advancement, its neighbors will get its energy which value is multiplied by a constant 0.98689. The lost energy is consumed to compromise the air resistance. And in this way, the whole rock fall will spread widely.

3.2.3 Zone 1

The reason that I introduce the first zone after the degree 45 slope zone is the 45 slope degree has the most integral rules and neighbors, that is, it is the most complicated zone.

As for the first zone of the mountain, the slope degree is 75 degree slope and above the 45 degree slope of the mountain. The neighbors of (0,0,0) are (0,-1,-2), (0,-1,-1), (1,-1,-1) and (-1,-1,-1).

In the real world, 75 degree slope is really precipitous, so when one of the four upper adjacent stones are moving and colliding with the lower stone. Then the lower stone will get different amounts of energy from them. The difference between it and 45 degree slope is that the second layer of above will also offer the stone energy because the slope is more precipitous than that of 45 degree. When it falls down, it will collide with the second layer below.

3.2.4 Zone 3

As for the third and the bottom zone of the mountain, the slope degree is 15 degree slope and under the 45 degree slope of the mountain. The neighbors of (0,0,0) are (0,-2,-1), (1,-2,-1) and (-1,-2,-1).

In the real world, 15 degree slope is really gradual, so when the stones are falling from the 45 degree slope and collided with the stone in this zone. Due to the difference of slope degree, the collided stone will start to bounce rather than to roll as that in the 45 degree zone. So in this way, the first collided stone will collide with next stone. One thing should be noticed that this stone is not adjacent to the first collided stone. We can say the process of rock falling in this zone is stone jumping not rolling. So in this zone, rule of rolling is not required.

4. Simulation results and analysis

After simulating our model in the new CD++ version, I obtain results (the .log file generated by the server) of three zones- 75, 45 and 15 in a (20, 20, 20) dimension of cells and analyze how the rock fall spreads from few falling stones. And I will also analyze some factors which will affect the process of rock fall, such as kinetic energy, resistance energy, and shape of each cell. Furthermore, we show the 3D results of the old CD++ version in order to see the general process of rock fall directly. Also, some advancement of the new version will be presented by the comparison between the old and new version. Finally, thanks to the addition of state variables and neighbor ports in the new version, it offers us more choices to make it happen. For example, we use the state variable (called stateVal) to record the state of the cell and it is used as one of conditions to determine which state the cell will change to. Detailed analysis will be discussed separately.

4.1 Shape=0 (all cells) and kinetic energy=500, 1000 and 8000 (special cells)

4.1.1 Rock fall on the slope with 75 degree

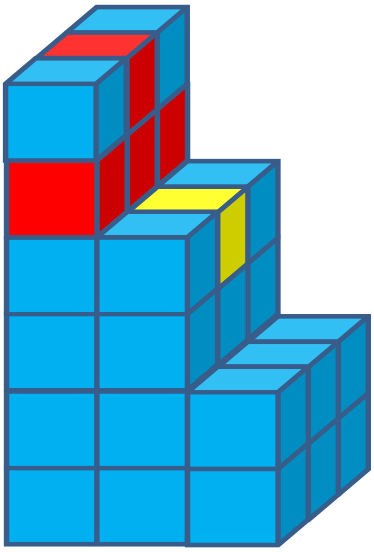

Figure 15. The cell and its neighbors in the zone 1

As can be seen from the figure 15, the yellow cell regarded as the cell will be triggered to fall when it obtains enough kinetic energy from its three neighbors (0,-1,-2), (0,-1,-1), (1,-1,-1) and (-1,-1,-1). It is reasonable that in the real world, the stone will be collided with three adjacent stones in the first layer of above and one in the second layer of above, because this slope is much steeper and the cell (0,-1,-2) will fall and collide with (0,0,0).

In the first case, we make all shapes of cells equal to 0, all state values and kinetic energy equal to 0 except for special cells, which are (4,2,4), (4,2,5), (4,4,8) and (4,10,14) with initial state, kinetic energy state variables (2, 500) respectively. The state variable value of friction of each cell is generated randomly ranging from 0-100.

```
0 / L / Y / 00:00:00:000:0 / fallrock(4,2,4)
(1646) / out /       0.00000 / fallrock(01)
0 / L / Y / 00:00:00:000:0 / fallrock(4,2,5)
(1647) / out /       0.00000 / fallrock(01)
0 / L / Y / 00:00:00:000:0 / fallrock(4,2,6)
(1648) / out /       0.00000 / fallrock(01)
```

Figure 16. Time 00:00:00:000:00 of (4,2,4) and (4,2,5)

At time 00:00:000:00, all cells load the initial cell values and output 0. And all cells are shown in order, like (4,2,4), (4,2,5) and (4,2,6).

```
0 / L / Y / 00:00:00:100:0 / fallrock(4,2,2)
(1644) / out /     -67.00000 / fallrock(01)
0 / L / Y / 00:00:00:100:0 / fallrock(4,2,3)
(1645) / out /     -47.00000 / fallrock(01)
0 / L / Y / 00:00:00:100:0 / fallrock(4,2,4)
(1646) / out /     493.94501 / fallrock(01)
0 / L / Y / 00:00:00:100:0 / fallrock(4,2,5)
(1647) / out /     493.94501 / fallrock(01)
0 / L / Y / 00:00:00:100:0 / fallrock(4,2,6)
(1648) / out /     -68.00000 / fallrock(01)
```

Figure 17. Time 00:00:00:100:0 of (4,2,4) and (4,2,5)

At time 00:00:100:00, specials cells start to output energy (493.94501) in the figure above, which is calculated by multiplying a physical constant 0.98689 plus a constant value 0.50. At the same time, other cells output negative values, such as (4,2,2) with -67.00000. 67 is the friction of it, and according to the rule defined before, so it will output (0-67.00000) = -67.00000. It means it is now in the motionless state. As explained before, motionless cells output negative values so they can be differentiated with toppling state. Also, the larger the negative value is, more kinetic energy will be required to make it fall. (That is, it is fixed well by adjacent stones.)

```
0 / L / Y / 00:00:00:200:0 / fallrock(3,3,5)
(1267) / out /     39.75241 / fallrock(01)
0 / L / Y / 00:00:00:200:0 / fallrock(3,4,8)
(1290) / out /    253.80355 / fallrock(01)
0 / L / Y / 00:00:00:200:0 / fallrock(3,5,9)
(1311) / out /      0.00000 / fallrock(01)
0 / L / Y / 00:00:00:200:0 / fallrock(3,12,15)
(1457) / out /    132.07356 / fallrock(01)
0 / L / Y / 00:00:00:200:0 / fallrock(4,3,5)
(1667) / out /    372.76541 / fallrock(01)
0 / L / Y / 00:00:00:200:0 / fallrock(4,3,6)
(1668) / out /    718.73861 / fallrock(01)
0 / L / Y / 00:00:00:200:0 / fallrock(4,5,9)
(1711) / out /    359.87050 / fallrock(01)
0 / L / Y / 00:00:00:200:0 / fallrock(4,12,15)
(1857) / out /    166.84413 / fallrock(01)
0 / L / Y / 00:00:00:200:0 / fallrock(5,3,6)
(2068) / out /      0.00000 / fallrock(01)
```

Figure 18. The collided cells in the first zone (3,3,5), (4,3,5), (4,3,6) and (5,3,6)

As time moves on, the neighbor cell obtains the kinetic energy from the special cells. As shown in the figure 13, now the yellow cell is (4,3,6) and the red cells in middle of the two upper layers are (4,2,4) and (4,2,5). It obtains the energy from both of the two special cells, so its energy is larger (718.73861) than the two values (493.94501) at the time 00:00:100:00. Also, (4,3,5) gets the energy only transferred by the cell (4,2,4), so the received energy is not that much, having 372.76541. One thing should be noticed is the received energy of each cell is already added by the other neighbors' resistance energy (negative value) which is represented by the state variable- friction. In the figure 16, it also shows energy outputs of cells another zone due to the order of showing log file. The details of the other two zones will be analyzed in 4.1.2 and 4.1.3 later.

The z axis of (4,3,6) is in the last layer of the first zone because the range of this zone is from (0,0,0) to (19,19,6). Energy of each cell will remain and not change any more. So at next time advancement no changes are in the each cell, they will not output their energy. They will not appear in the log file.

```
0 / L / Y / 00:00:00:300:0 / fallrock(2,4,8)
(890) / out /     42.88771 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(3,3,5)
(1267) / out /     20.24506 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(3,4,6)
(1288) / out /      0.00000 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(3,4,8)
(1290) / out /    126.56444 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(3,5,9)
(1311) / out /    585.71514 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(3,12,15)
(1457) / out /     66.10105 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(3,14,16)
(1498) / out /      0.00000 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(4,3,5)
(1667) / out /    185.65291 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(4,3,6)
(1668) / out /    357.49810 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(4,4,6)
(1688) / out /    228.22563 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(4,5,9)
(1711) / out /    179.24799 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(4,6,10)
(1732) / out /     95.99709 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(4,12,15)
(1857) / out /     83.37163 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(4,14,16)
(1898) / out /    392.54896 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(5,4,8)
(2090) / out /    125.26175 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(5,5,9)
(2111) / out /    582.27514 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(5,12,15)
(2257) / out /    130.24890 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(5,14,16)
(2298) / out /    166.96387 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(6,4,8)
(2490) / out /     59.58429 / fallrock(01)
```

Figure 19. Time 00:00:00:300:0 (energy=500)

```
0 / L / Y / 00:00:00:300:0 / fallrock(3,12,15)
(1457) / out /    309.58903 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(3,14,16)
(1498) / out /   1057.36192 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(4,3,5)
(1667) / out /    429.14088 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(4,3,6)
(1668) / out /    795.77646 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(4,4,6)
(1688) / out /    936.11812 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(4,5,9)
(1711) / out /    422.73597 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(4,6,10)
(1732) / out /    803.21478 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(4,12,15)
(1857) / out /    326.85960 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(4,14,16)
(1898) / out /   1853.54043 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(5,3,5)
(2067) / out /     87.85922 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(5,3,6)
(2068) / out /    130.66064 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(5,4,6)
(2088) / out /    457.60843 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(5,4,8)
(2090) / out /    310.31261 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(5,5,9)
(2111) / out /    116.27178 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(5,6,10)
(2132) / out /    267.85140 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(5,12,15)
(2257) / out /    373.73688 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(5,14,16)
(2298) / out /   1140.95818 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(6,4,8)
(2490) / out /    340.87384 / fallrock(01)
```

Figure 20. Time 00:00:00:300:0 (energy=1000)

Based on the comparison of figure 19 and figure 20 above, we can see that the same cell in the same zone has more energy when the initial cell given more initial kinetic energy. Also, because of more energy can be transferred between cells, more cells will be triggered to fall, such as (5,3,6) in the second case (with more initial energy-1000). By contrast, in the first case, the cell (5,3,6) could not get enough kinetic energy to fall due to the total of neighbors' resistance larger than received energy.

```
0 / L / Y / 00:00:01:400:0 / fallrock(16,8,12)
(6574) / out /    5398.32997 / fallrock(01)
0 / L / Y / 00:00:01:400:0 / fallrock(17,6,10)
(6932) / out /    2313.33053 / fallrock(01)
0 / L / Y / 00:00:01:400:0 / fallrock(17,7,11)
(6953) / out /    6990.90159 / fallrock(01)
0 / L / Y / 00:00:01:400:0 / fallrock(17,8,12)
(6974) / out /   11582.39154 / fallrock(01)
0 / L / Y / 00:00:01:500:0 / fallrock(17,6,10)
(6932) / out /    1149.53331 / fallrock(01)
0 / L / Y / 00:00:01:500:0 / fallrock(17,7,11)
(6953) / out /    3472.88696 / fallrock(01)
0 / L / Y / 00:00:01:500:0 / fallrock(17,8,12)
(6974) / out /    5753.48406 / fallrock(01)
0 / L / Y / 00:00:01:500:0 / fallrock(18,6,10)
(7332) / out /    1642.51348 / fallrock(01)
0 / L / Y / 00:00:01:500:0 / fallrock(18,7,11)
(7353) / out /    6271.83412 / fallrock(01)
0 / L / Y / 00:00:01:500:0 / fallrock(18,8,12)
(7374) / out /   11900.90718 / fallrock(01)
0 / L / Y / 00:00:01:600:0 / fallrock(18,6,10)
(7332) / out /     816.33789 / fallrock(01)
0 / L / Y / 00:00:01:600:0 / fallrock(18,7,11)
(7353) / out /    3115.72552 / fallrock(01)
0 / L / Y / 00:00:01:600:0 / fallrock(18,8,12)
(7374) / out /    5911.69105 / fallrock(01)
```

Figure21. 00:00:01:600:0(energy=8000)

In this case, we assign the initial cells with more energy in order to ensure all the cells do the transition at least one time according to the rules at the overtime 00:00:01:600:0. In the real world, it means that most of parts of the whole mountain crush.

4.1.2 Rock fall on the slope with 45 degree

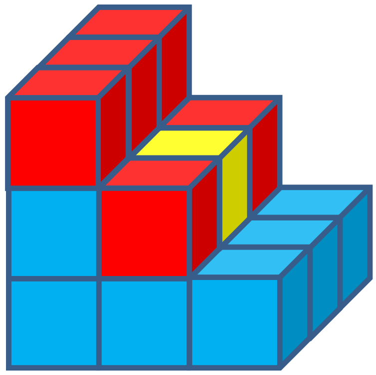

Figure 22. The cell and its neighbors in the zone 2

As can be seen from the figure 22, the yellow cell regarded as the cell will be triggered to fall when it obtains enough kinetic energy from its five neighbors (0,-1,-1), (1,-1,-1), (-1,-1,-1), (1,0,0) and (-1,0,0). This zone is different from the upper zone with two more neighbors, so more cells can be triggered.

This zone is ranging from (0,0,8) to (19,19,12). It is important that one layer should be left empty after several times tries to make 75 and 45 degree zone connect well. The reason is that due to the 'nowrapped' characteristics of our model, cells in the first layer of the 45 zone tries to find its neighbors in upper layer but they belong to the first zone, as a result, invalid neighbor exception will be thrown. So we define the second zone ranges starting from (0,0,8) rather than from (0,0,7).

```
0 / L / Y / 00:00:00:300:0 / fallrock(2,4,8)
(890) / out /      42.88771 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(3,3,5)
(1267) / out /      20.24506 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(3,4,6)
(1288) / out /       0.00000 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(3,4,8)
(1290) / out /     126.56444 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(3,5,9)
(1311) / out /     585.71514 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(3,12,15)
(1457) / out /      66.10105 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(3,14,16)
(1498) / out /       0.00000 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(4,3,5)
(1667) / out /     185.65291 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(4,3,6)
(1668) / out /     357.49810 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(4,4,6)
(1688) / out /     228.22563 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(4,5,9)
(1711) / out /     179.24799 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(4,6,10)
(1732) / out /      95.99709 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(4,12,15)
(1857) / out /      83.37163 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(4,14,16)
(1898) / out /     392.54896 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(5,4,8)
(2090) / out /     125.26175 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(5,5,9)
(2111) / out /     582.27514 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(5,12,15)
(2257) / out /     130.24890 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(5,14,16)
(2298) / out /     166.96387 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(6,4,8)
(2490) / out /      59.58429 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(6,14,16)
(2698) / out /       0.00000 / fallrock(01)
```

Figure 23. All changed cells at time 00:00:00:300:0

The result is the same as we expected. The number of changed cells in this zone is 8, while that in the zone 75 and 15 is 7 and 5, respectively. As the energy is not large, so the difference is not so much. If the initial energy is much larger, the result will be more apparent.

4.1.3 Rock fall on the slope with 15 degree

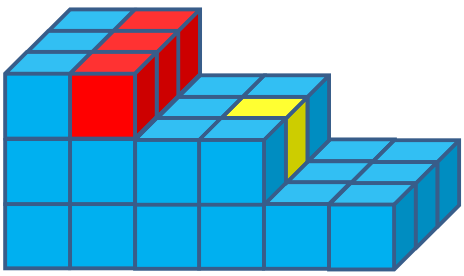

Figure 24. ***The cell and its neighbors in the zone 3***

As can be seen from the figure 24, the yellow cell regarded as the cell will be triggered to fall when it obtains enough kinetic energy from its three neighbors (0,-2,-1), (1,-2,-1), (-1,-2,-1),. This zone is different from the upper zone with fewer neighbors. And the cell will bounce rather than roll when it is triggered.

```
0 / L / Y / 00:00:00:400:0 / fallrock(2,4,8)
(890) / out /      21.80236 / fallrock(01)
0 / L / Y / 00:00:00:400:0 / fallrock(2,5,9)
(911) / out /     481.07659 / fallrock(01)
0 / L / Y / 00:00:00:400:0 / fallrock(3,5,9)
(1311) / out /     291.42522 / fallrock(01)
0 / L / Y / 00:00:00:400:0 / fallrock(3,6,10)
(1332) / out /     611.61144 / fallrock(01)
0 / L / Y / 00:00:00:400:0 / fallrock(3,16,17)
(1539) / out /     258.84106 / fallrock(01)
0 / L / Y / 00:00:00:400:0 / fallrock(4,4,6)
(1688) / out /     113.85987 / fallrock(01)
0 / L / Y / 00:00:00:400:0 / fallrock(4,6,10)
(1732) / out /      48.18184 / fallrock(01)
0 / L / Y / 00:00:00:400:0 / fallrock(4,14,16)
(1898) / out /     195.47941 / fallrock(01)
0 / L / Y / 00:00:00:400:0 / fallrock(4,16,17)
(1939) / out /     481.33685 / fallrock(01)
0 / L / Y / 00:00:00:400:0 / fallrock(5,5,9)
(2111) / out /     289.71657 / fallrock(01)
0 / L / Y / 00:00:00:400:0 / fallrock(5,6,10)
(2132) / out /     444.85176 / fallrock(01)
0 / L / Y / 00:00:00:400:0 / fallrock(5,14,16)
(2298) / out /      83.43110 / fallrock(01)
0 / L / Y / 00:00:00:400:0 / fallrock(5,16,17)
(2339) / out /     411.79570 / fallrock(01)
0 / L / Y / 00:00:00:400:0 / fallrock(6,4,8)
(2490) / out /      30.09557 / fallrock(01)
0 / L / Y / 00:00:00:400:0 / fallrock(6,5,9)
(2511) / out /     312.94889 / fallrock(01)
0 / L / Y / 00:00:00:400:0 / fallrock(6,16,17)
(2739) / out /       0.00000 / fallrock(01)
```

Figure 25. At time 00:00:00:400:0

In the above figure, we can see that in this zone the cell is jumping to transfer energy as we explained in the 3.2, such as the cell (4,14,16), (4,16,17), (5,16,17) and (6,16,17). The y axis of all of them is even not odd. Because the first cell value we assign in the stavalues.file is (4,10,14), the cell only transfer its energy to cells belonging to the y axis (y+1). Also the energy in the (4,14,16) is 195.47941 and that in the (4,14,17) is 481.33685 in the the lower layer. More kinetic energy here means this stone has a greater speed. But this energy is transferred by the left and right neighbor of (4,14,16) not itself. An interesting cell in this figure is (6,16,17), its output is 0.00000 (means it is in the toppling state), because received energy is just larger than its resistance and smaller than the value of resistance plus 20. So it topples when it is collided with other stones but it will not begin to fall.

4.2 Effects of different values of state variable- friction

```
0 / L / Y / 00:00:00:200:0 / fallrock(3,3,5)
(1267) / out /      66.73438 / fallrock(01)
0 / L / Y / 00:00:00:200:0 / fallrock(3,3,6)
(1268) / out /      82.17251 / fallrock(01)
0 / L / Y / 00:00:00:200:0 / fallrock(3,4,8)
(1290) / out /     357.75763 / fallrock(01)
0 / L / Y / 00:00:00:200:0 / fallrock(3,5,9)
(1311) / out /     114.32040 / fallrock(01)
0 / L / Y / 00:00:00:200:0 / fallrock(3,12,15)
(1457) / out /     384.40859 / fallrock(01)
0 / L / Y / 00:00:00:200:0 / fallrock(4,3,5)
(1667) / out /     355.39999 / fallrock(01)
0 / L / Y / 00:00:00:200:0 / fallrock(4,3,6)
(1668) / out /     857.92026 / fallrock(01)
0 / L / Y / 00:00:00:200:0 / fallrock(4,5,9)
(1711) / out /     401.23755 / fallrock(01)
0 / L / Y / 00:00:00:200:0 / fallrock(4,12,15)
(1857) / out /     315.86088 / fallrock(01)
0 / L / Y / 00:00:00:200:0 / fallrock(5,3,5)
(2067) / out /     169.51620 / fallrock(01)
0 / L / Y / 00:00:00:200:0 / fallrock(5,3,6)
(2068) / out /     172.47666 / fallrock(01)
0 / L / Y / 00:00:00:200:0 / fallrock(5,4,8)
(2090) / out /     309.07883 / fallrock(01)
0 / L / Y / 00:00:00:200:0 / fallrock(5,5,9)
(2111) / out /     110.64465 / fallrock(01)
0 / L / Y / 00:00:00:200:0 / fallrock(5,12,15)
(2257) / out /     425.13983 / fallrock(01)
```

Figure 26. At time 00:00:00:200:0 with random friction from 0-50

```
0 / L / Y / 00:00:00:200:0 / fallrock(3,3,5)
(1267) / out /      39.75241 / fallrock(01)
0 / L / Y / 00:00:00:200:0 / fallrock(3,4,8)
(1290) / out /     253.80355 / fallrock(01)
0 / L / Y / 00:00:00:200:0 / fallrock(3,5,9)
(1311) / out /       0.00000 / fallrock(01)
0 / L / Y / 00:00:00:200:0 / fallrock(3,12,15)
(1457) / out /     132.07356 / fallrock(01)
0 / L / Y / 00:00:00:200:0 / fallrock(4,3,5)
(1667) / out /     372.76541 / fallrock(01)
0 / L / Y / 00:00:00:200:0 / fallrock(4,3,6)
(1668) / out /     718.73861 / fallrock(01)
0 / L / Y / 00:00:00:200:0 / fallrock(4,5,9)
(1711) / out /     359.87050 / fallrock(01)
0 / L / Y / 00:00:00:200:0 / fallrock(4,12,15)
(1857) / out /     166.84413 / fallrock(01)
0 / L / Y / 00:00:00:200:0 / fallrock(5,3,6)
(2068) / out /       0.00000 / fallrock(01)
0 / L / Y / 00:00:00:200:0 / fallrock(5,4,8)
(2090) / out /     251.18085 / fallrock(01)
0 / L / Y / 00:00:00:200:0 / fallrock(5,5,9)
(2111) / out /       0.00000 / fallrock(01)
0 / L / Y / 00:00:00:200:0 / fallrock(5,12,15)
(2257) / out /     261.22141 / fallrock(01)
```

Figure 27. At time 00:00:00:200:0 with random friction from 0-100

Through the analysis and comparison between the figure 26 and 27, we can see that more cells are triggered to fall with less friction resistance. For example, the cell (3,3,6) and (5,3,5) starts to output energy at this time, while it is not shown in the figure 27.

```
0 / L / Y / 00:00:00:700:0 / fallrock(1,7,11)
(553) / out /     383.49769 / fallrock(01)
0 / L / Y / 00:00:00:700:0 / fallrock(1,8,12)
(574) / out /     101.25289 / fallrock(01)
0 / L / Y / 00:00:00:700:0 / fallrock(2,8,12)
(974) / out /     427.71692 / fallrock(01)
0 / L / Y / 00:00:00:700:0 / fallrock(6,8,12)
(2574) / out /     217.47095 / fallrock(01)
0 / L / Y / 00:00:00:700:0 / fallrock(7,7,11)
(2953) / out /     134.85721 / fallrock(01)
0 / L / Y / 00:00:00:700:0 / fallrock(7,8,12)
(2974) / out /     533.03153 / fallrock(01)
0 / L / Y / 00:00:00:700:0 / fallrock(8,7,11)
(3353) / out /     165.46190 / fallrock(01)
0 / L / Y / 00:00:00:800:0 / fallrock(7,8,12)
(2974) / out /     265.25723 / fallrock(01)
0 / L / Y / 00:00:00:800:0 / fallrock(8,7,11)
(3353) / out /      82.68507 / fallrock(01)
0 / L / Y / 00:00:00:800:0 / fallrock(8,8,12)
(3374) / out /     513.01517 / fallrock(01)
0 / L / Y / 00:00:00:800:0 / fallrock(9,7,11)
(3753) / out /       0.00000 / fallrock(01)
0 / L / Y / 00:00:00:900:0 / fallrock(8,8,12)
(3374) / out /     255.31509 / fallrock(01)
0 / L / Y / 00:00:00:900:0 / fallrock(9,8,12)
(3774) / out /     372.82666 / fallrock(01)
0 / L / Y / 00:00:01:000:0 / fallrock(9,8,12)
(3774) / out /     185.68333 / fallrock(01)
0 / L / Y / 00:00:01:000:0 / fallrock(10,8,12)
(4174) / out /     208.55218 / fallrock(01)
0 / L / Y / 00:00:01:100:0 / fallrock(10,8,12)
(4174) / out /     104.08805 / fallrock(01)
0 / L / Y / 00:00:01:100:0 / fallrock(11,8,12)
(4574) / out /      74.54777 / fallrock(01)
0 / L / Y / 00:00:01:200:0 / fallrock(11,8,12)
(4574) / out /      37.52794 / fallrock(01)
```

Figure 28. At the time end with random friction from 0-50

```
0 / L / Y / 00:00:00:700:0 / fallrock(1,6,10)
(532) / out /     412.47550 / fallrock(01)
0 / L / Y / 00:00:00:700:0 / fallrock(1,7,11)
(553) / out /      31.88919 / fallrock(01)
0 / L / Y / 00:00:00:700:0 / fallrock(1,8,12)
(574) / out /     377.97360 / fallrock(01)
0 / L / Y / 00:00:00:700:0 / fallrock(2,7,11)
(953) / out /     537.79236 / fallrock(01)
0 / L / Y / 00:00:00:700:0 / fallrock(2,8,12)
(974) / out /    1342.29835 / fallrock(01)
0 / L / Y / 00:00:00:700:0 / fallrock(3,8,12)
(1374) / out /     199.25656 / fallrock(01)
0 / L / Y / 00:00:00:700:0 / fallrock(4,8,12)
(1774) / out /     222.17737 / fallrock(01)
0 / L / Y / 00:00:00:700:0 / fallrock(5,8,12)
(2174) / out /     172.10050 / fallrock(01)
0 / L / Y / 00:00:00:700:0 / fallrock(6,8,12)
(2574) / out /      39.55046 / fallrock(01)
0 / L / Y / 00:00:00:700:0 / fallrock(7,6,10)
(2932) / out /     130.24423 / fallrock(01)
0 / L / Y / 00:00:00:700:0 / fallrock(7,7,11)
(2953) / out /      73.98711 / fallrock(01)
0 / L / Y / 00:00:00:700:0 / fallrock(8,7,11)
(3353) / out /      83.25054 / fallrock(01)
0 / L / Y / 00:00:00:800:0 / fallrock(1,8,12)
(574) / out /     188.23982 / fallrock(01)
0 / L / Y / 00:00:00:800:0 / fallrock(2,8,12)
(974) / out /     667.22077 / fallrock(01)
0 / L / Y / 00:00:00:800:0 / fallrock(8,7,11)
(3353) / out /      41.85062 / fallrock(01)
```

Figure 29. At the time end with random friction from 0-100

As can be seen from the figure 28 and 29, at the time end of the .log life, the time ends at 00:00:01:200:0 in the first case (random friction from 0 to 50) while that ends at 00:00:00:800:0 in the second one (from 0- 100). We can conclude that with less friction resistance, during the process of fall rock, consume of energy is less, so it will last longer.

4.3 Effects of different values of state variable- shape

```
0 / L / Y / 00:00:00:300:0 / fallrock(3,6,10)
(1332) / out /   6827.65867 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(3,12,15)
(1457) / out /   3785.52923 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(3,14,16)
(1498) / out /  15118.09327 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(4,3,5)
(1667) / out /      0.00000 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(4,3,6)
(1668) / out /      0.00000 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(4,4,6)
(1688) / out /  10570.28943 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(4,5,9)
(1711) / out /      0.00000 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(4,6,10)
(1732) / out /  10649.31664 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(4,12,15)
(1857) / out /   3848.19674 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(4,14,16)
(1898) / out /  22674.00292 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(5,3,5)
(2067) / out /      0.00000 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(5,4,6)
(2088) / out /   6639.10329 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(5,4,8)
(2090) / out /      0.00000 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(5,5,9)
(2111) / out /      0.00000 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(5,6,10)
(2132) / out /   6710.68178 / fallrock(01)
```

Figure 30. At the time 00:00:00:300:0 with cells have more shape 1 than 0

Figure 30 shows the simulation result at time 00:00:00:300:0 with the same state variable value except for 'shape'. We generate the value of shape randomly but ensure that more 1 than 0 and assign them to each cell. As what we explained before, 0 represents the shape of this cell is regular, while 1 means irregular.

```
0 / L / Y / 00:00:00:300:0 / fallrock(3,6,10)
(1332) / out /   6790.11014 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(3,12,15)
(1457) / out /   3821.55071 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(3,14,16)
(1498) / out /  14950.72202 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(4,3,5)
(1667) / out /   3757.81735 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(4,3,6)
(1668) / out /   6892.71110 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(4,4,6)
(1688) / out /  10511.60261 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(4,5,9)
(1711) / out /      0.00000 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(4,6,10)
(1732) / out /  10579.48317 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(4,12,15)
(1857) / out /   3772.69966 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(4,14,16)
(1898) / out /  22825.21205 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(5,3,5)
(2067) / out /   1708.08882 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(5,3,6)
(2068) / out /   1677.95907 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(5,4,6)
(2088) / out /   6726.84799 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(5,4,8)
(2090) / out /      0.00000 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(5,5,9)
(2111) / out /   1649.57611 / fallrock(01)
0 / L / Y / 00:00:00:300:0 / fallrock(5,6,10)
(2132) / out /   6561.72933 / fallrock(01)
```

Figure 31. At the time 00:00:00:300:0 with cells have more shape 0 than 1

According to the same method, we generate the value of shape randomly but ensure that more 0 than 1 and assign them to each cell.

Through comparison, it is obvious that in the figure 30, many cells are outputting 0.00000 which means they are in the state- toppling. The reason is that when the stone is collided with other stone, its state value changes from 0 (motionless) to 2 (falling). And at the next time advancement, if the shape is 1 and the state value is 2, the state value will change to 4 (bouncing) and output 0.

It can make sense well in the real world. When a stone is collided with another stone and begin to fall down from the mountain, it should bounce to affect the stone in the lower layer rather than the one adjacent to it if its shape is irregular.

5. Conclusion

We have used the new CD++ version to simulate our rock fall model with (20,20,20) dimension in

3D dimension and analyzed the result of the whole process of simulation.

Firstly, we have different zones which represent different slops of the mountain. And they present different behaviors of rock fall because the cell in different zones has different neighbors and differing rules. The 15 degree, the (0,0,0) will only obtain energy from three neighbors from the upper layer, while 75 and 45 degree zone receive energy from 4 and 5 neighbors in the two layers of the above and left and right and the upper layers, respectively.

Secondly, we compared and analyzed the effect of state variable- kinetic energy. The larger kinetic energy assigned to the starting cells, more cells will be triggered to fall. The end time will last long and trigger all the cells to make a transition according to the rule at least one time except for some boundary ones.

Thirdly, we analyzed the effect of state variable- friction. We randomly generated the friction value of each cell ranging from 0- 50 and 0- 100 respectively and assigned them to each cell. After analysis, the friction from 0- 50 can make more cells change states and output their energy. Specifically, the process of rock fall will make more stones involved.

Finally, we analyzed the effect of state variable- shape. The more values of shape 1 are, the more cells output value 0.00000 (means the state- toppling) with 0 kinetic energy. The more values of shape 0 are, the more cells output its kinetic energy.

6. References

[1] G. Wainer: "CD++: a Toolkit to Define Discrete-Event Models", *Software, Practice and Experience*, Wiley, Vol. 32, No 3. pp. 1261-1306. November 2002.

[2] C. Phillips: "A Review of High Performance Simulation Tools and Modeling Concepts", *Recent Advances in Modeling and Simulation Tools for Communication Networks and Services*, pp. 29-47, 2007

[3] G. Wainer, N. Giambiasi: "Application of the Cell-DEVS Paradigm for Cell Spaces Modeling and Simulation", *Simulation*, Vol. 71, No. 1, pp. 22-39, January 2001.

[4] F. Guzzetti, G. Crosta, R. Detti and F. Agliardi, STONE: "a computer program for the three-dimensional simulation of rock-falls", *Computers & Geosciences*, Volume 28, Issue 9, Pages 1079-1093, November 2002

Appendix

The rule of the FallRockV2 model:

```
[top]
components : FallRock

[FallRock]
type : cell
dim : (20,20,20)
delay : transport
defaultDelayTime : 100
border : nowrapped
neighbors : FallRock(0,-1,-2)
neighbors : FallRock(1,-1,-1) FallRock(0,-1,-1)
FallRock(-1,-1,-1)
neighbors : FallRock(1,0,0)    FallRock(0,0,0)
FallRock(-1,0,0)
neighbors : FallRock(1,1,1)    FallRock(0,1,1)
FallRock(-1,1,1)
neighbors : FallRock(0,-2,-1) FallRock(1,-2,-1)
FallRock(-1,-2,-1)
initialValue : 0
initialCellsValue : init.val
StateVariables : stateVal kineticEnergyVal
frictionVal shapeVal
stateValues : 0 0 0 0
InitialVariablesValue : initial.stvalues
zone : slope75 { (0,0,0)..(19,19,6) }
zone : slope45 { (0,0,8)..(19,19,12) }
zone : slope15 { (0,0,14)..(19,19,19) }
localTransition : FallRockBehavior

[slope75]
%Rock Falls along 75 degree

%Motionless
rule :
{
        ( 0 - $frictionVal )
}
{
        $stateVal := 0 ;
        $kineticEnergyVal := 0 ;
}


100
{
             (( 0.98689 * $kineticEnergyVal
                        + 0.40 * (0,-1,-2)
                        + 0.50 * (0,-1,-1)
                        + 0.23 * (1,-1,-1)
                        + 0.23 * (-1,-1,-1)
                        ) < $frictionVal
               ) AND
               (
                $stateVal = 0
               )
}

%Toppling
rule :
{
  0
}
{
        $stateVal := 1 ;
        $kineticEnergyVal := 0 ;

}
100
{
             (( 0.98689 * $kineticEnergyVal
                        + 0.40 * (0,-1,-2)
                        + 0.50 * (0,-1,-1)
                        + 0.23 * (1,-1,-1)
                        + 0.23 * (-1,-1,-1)
                        ) > $frictionVal
               ) AND
               (( 0.98689 * $kineticEnergyVal
                        + 0.40 * (0,-1,-2)
                        + 0.50 * (0,-1,-1)
                        + 0.23 * (1,-1,-1)
                        + 0.23 * (-1,-1,-1)
                        ) < ($frictionVal+20)
```

```
        ) AND
        (
         $stateVal = 0     OR   $stateVal = 1
        )
}



%falling
rule :
{

        (( 0.98689 * $kineticEnergyVal
            + 0.40 * (0,-1,-2)
            + 0.50 * (0,-1,-1)
            + 0.23 * (1,-1,-1)
            + 0.23 * (-1,-1,-1)
         )-   $frictionVal)

}
{
    $stateVal := 2 ;
    $kineticEnergyVal :=
            (( 0.98689 * $kineticEnergyVal
            + 0.40 * (0,-1,-2)
            + 0.50 * (0,-1,-1)
            + 0.23 * (1,-1,-1)
            + 0.23 * (-1,-1,-1)
             )-   $frictionVal) ;

}
100
{
       (( 0.98689 * $kineticEnergyVal
            + 0.40 * (0,-1,-2)
            + 0.50 * (0,-1,-1)
            + 0.23 * (1,-1,-1)
            + 0.23 * (-1,-1,-1)
             ) > ($frictionVal + 20)
        ) AND
        (
         $stateVal = 0     OR   $stateVal = 1
        )
```

```
}
 %rolling

rule :
{
    ( 0.98689 * $kineticEnergyVal + 0.50 )
}
{
    $stateVal := 3 ;
}
100
{

        $stateVal = 2 AND
        $shapeVal = 0

}

%bouncing
rule :
{
    0
}
{
    $stateVal := 4 ;
}
100
{

        $stateVal = 2     AND
       $shapeVal = 1

}
rule :
{ ( 0,0,0 ) } 0 { t }

[slope45]

%Rock Falls along 45 degree

%Motionless
rule :
```

```
{
        ( 0 - $frictionVal )
}
{
        $stateVal := 0 ;
        $kineticEnergyVal := 0 ;
}
100
{
             (( 0.98689 * $kineticEnergyVal
                        + 0.23 * (1,-1,-1)
                        + 0.50 * (0,-1,-1)
                        + 0.23 * (-1,-1,-1)
                        + 0.38 * (1,0,0)
                        + 0.38 * (-1,0,0)
                          ) < $frictionVal
               ) AND
               (
                $stateVal = 0
               )
}

%Toppling
rule :
{
  0
}
{
        $stateVal := 1 ;
        $kineticEnergyVal := 0 ;

}
100
{
             (( 0.98689 * $kineticEnergyVal
                        + 0.23 * (1,-1,-1)
                        + 0.50 * (0,-1,-1)
                        + 0.23 * (-1,-1,-1)
                        + 0.38 * (1,0,0)
                        + 0.38 * (-1,0,0)
                          ) > $frictionVal
               ) AND
               (( 0.98689 * $kineticEnergyVal
                        + 0.23 * (1,-1,-1)
                        + 0.50 * (0,-1,-1)
                        + 0.23 * (-1,-1,-1)
                        + 0.38 * (1,0,0)
                        + 0.38 * (-1,0,0)
                          ) < ($frictionVal+20)
               ) AND
               (
                $stateVal = 0      OR   $stateVal = 1
               )
}

%falling
rule :
{

                (( 0.98689 * $kineticEnergyVal
                + 0.23 * (1,-1,-1)
                + 0.50 * (0,-1,-1)
                + 0.23 * (-1,-1,-1)
                + 0.38 * (1,0,0)
                + 0.38 * (-1,0,0)
                + 0.50 )-    $frictionVal)

}
{
        $stateVal := 2 ;
        $kineticEnergyVal :=
                        (( 0.98689 * $kineticEnergyVal
                        + 0.23 * (1,-1,-1)
                        + 0.50 * (0,-1,-1)
                        + 0.23 * (-1,-1,-1)
                        + 0.38 * (1,0,0)
                        + 0.38 * (-1,0,0)
                        + 0.50 )-    $frictionVal) ;

}
100
{
             (( 0.98689 * $kineticEnergyVal
                        + 0.23 * (1,-1,-1)
                        + 0.50 * (0,-1,-1)
                        + 0.23 * (-1,-1,-1)
```

```
                        + 0.38 * (1,0,0)
                        + 0.38 * (-1,0,0)
                        + 0.50 ) > ($frictionVal + 20)
                ) AND
                (
                 $stateVal = 0      OR   $stateVal = 1
                )

}

  %rolling

rule :
{
        ( 0.98689 * $kineticEnergyVal + 0.50 )
}
{
        $stateVal := 3 ;
}
100
{

                 $stateVal = 2    AND
          $shapeVal = 0

}

%bouncing
rule :
{
        0
}
{
        $stateVal := 4 ;
}
100
{

                 $stateVal = 2 AND
          $shapeVal = 1

}
rule :
```

```
{ ( 0,0,0 ) } 0 { t }

[slope15]

%Rock Falls along 15 degree
%Motionless
rule :
{
        ( 0 - $frictionVal )
}
{
        $stateVal := 0 ;
        $kineticEnergyVal := 0 ;
}
100
{
             (( 0.98689 * $kineticEnergyVal
                        + 0.50 * (0,-2,-1)
                        + 0.50 * (1,-2,-1)
                        + 0.50 * (-1,-2,-1)
                        + 0.50 ) < $frictionVal
               ) AND
               (
                $stateVal = 0
               )
}


%Toppling
rule :
{
  0
}
{
        $stateVal := 1 ;
        $kineticEnergyVal := 0 ;

}
100
{
             (( 0.98689 * $kineticEnergyVal
                        + 0.50 * (0,-2,-1)
                        + 0.50 * (1,-2,-1)
```

```
            + 0.50 * (-1,-2,-1)
            + 0.50 ) > $frictionVal
        ) AND
        (( 0.98689 * $kineticEnergyVal
            + 0.50 * (0,-2,-1)
            + 0.50 * (1,-2,-1)
            + 0.50 * (-1,-2,-1)
            + 0.50 ) < ($frictionVal+50)
        ) AND
        (
         $stateVal = 0      OR   $stateVal = 1
        )
}



%falling
rule :
{

        (( 0.98689 * $kineticEnergyVal
            + 0.50 * (0,-2,-1)
            + 0.50 * (1,-2,-1)
            + 0.50 * (-1,-2,-1)
        + 0.50 )-    $frictionVal)

}
{
    $stateVal := 2 ;
    $kineticEnergyVal :=
            (( 0.98689 * $kineticEnergyVal
            + 0.50 * (0,-2,-1)
            + 0.50 * (1,-2,-1)
            + 0.50 * (-1,-2,-1)
            + 0.50 )-    $frictionVal) ;

}
100
{
       (( 0.98689 * $kineticEnergyVal
            + 0.50 * (0,-2,-1)
            + 0.50 * (1,-2,-1)
            + 0.50 * (-1,-2,-1)
            + 0.50 ) > ($frictionVal + 50)
        ) AND
        (
         $stateVal = 0      OR   $stateVal = 1
        )

}


%bouncing
rule :
{
    ( 0.98689 * $kineticEnergyVal + 0.50 )
}
{
    $stateVal := 4 ;
}
100
{

        $stateVal = 2      OR         $stateVal
= 3
}
rule :
{ ( 0,0,0 ) } 0 { t }


[FallRockBehavior]

%other cells behavior

rule :
{ ( 0,0,0 ) } 0 { t }
```