

2D Crystal Growth Simulation with Cell-DEVS based on extension CD++

Weiwei Li

Department of Electronics and Computer Engineering

University of Ottawa

161 Louis Pasteur Pvt

Ottawa, ON K1N 6N5 Canada

Wli086@uottawa.ca

ABSTRACT: *Cellular automata have been widely used to model and simulate different physical phenomenon. Cell-DEVS is an extension of DEVS. With the help of Cell-DEVS, I could simulate and study the group behaviors as it supports asynchronous evaluation. In this paper, I use extension CD++ (a toolkit for Cell-DEVS) to simulate a two dimension crystal growth. In this simulation, I study the impact of different temperatures, different concentration of impurities and defects as well as gravity. Simulation results show that the temperature and concentration of impurities are the key factors that affect the physical size of crystal. High concentration of impurities will result the crystal stop growing, which is called dead zone effect. High concentration of defects will also affect the size of crystal. However, it does not cause the crystal to stop growing. Both the impurities and defects have a significant affection on the qualities of the crystal but with different ways. The impurities will form a cluster and the crystal orientations of different atoms inside the cluster are random. The defects will form dislocation while around the dislocation the crystal orientation could still be same. These results show that the impurities will have a much more negative effect on the qualities of crystal than the defects. Gravity will have a tiny effect on the growth on the crystal. However, it will affect the crystal orientation of the crystal, especially in some extreme circumstance where the gravity is very high, the affection of gravity is significant.*

Key Words: CD++, Cell-DEVS, Crystal Growth

1. Introduction

With the development of simulation tools, we could be able to simulate much more complex system as well as physical phenomenon. Simulation could not only reduce the cost but also provide a way to study the system which is not proper to do experiments. There are a lot of different techniques used in simulation and Discrete Event System Specification (DEVS) is one of the widely used techniques nowadays. The key concept for DEVS is to divide the system into atomic models and then combine these atomic models to form coupled models. As a result, any complex system could be divided into coupled models and then further divided into atomic models. Besides, as DEVS is event-driven, it could significantly reduce the simulation overhead. There are a lot of different extensions of DEVS and one among them is Cell-DEVS. Cell-DEVS supports cellular automata and each cell could have different time

delays. With the assistance of Cell-DEVS, we could simulate group behaviors and a lot of successful examples using Cell-DEVS are presented [1]. In this paper, we will use a toolkit of Cell-DEVS which is called CD++ [2] to simulate the crystal growth and some newly introduced feature of CD++ will be used to improve the model. We will discuss the impacts of temperature, impurities, defects and gravity in details.

2. Background

2.1. CD++ specification

2.1.1. DEVS specification

The toolkit of Cell-DEVS CD++ is implemented by C++ and thus is fully compatible with C++. CD++ has implemented a simulation framework using algorithm pattern [3]. Thus, the user just needs to implement a few functions within an

atomic model to make it work. As a result, even people with little programming knowledge could still build a complicated model. The functions that the user needs to implement within an atomic model of CD++ are external function, internal function, output function and initial function. To make different atomic models work together, user needs to write configuration file (called ma file in CD++) to define the input ports, output ports and links between the input ports and output ports of different atomic models as well as couple models. User could also define a select function deciding which atomic model goes first if several atomic models are activated by a same event.

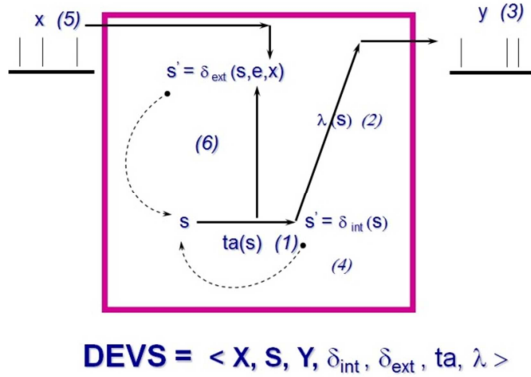


Figure 2.1. Informal DEVS definition of atomic model [4]

Figure 2.1 shows an informal definition of atomic model. There is an input port x and output port y . s is the state of the atomic model. ta is the time advance function which means after the time advance is elapsed, output function and internal function will be executed. Although by logic, output function and internal function should happen at the same time, it is always recommended to consider output function runs before internal function. This is because internal function might change the state of the atomic model and if internal function is executed first and then followed by the output function, the output value might be wrong. External function handles the events coming from the input port. External function might change the state of the atomic model but it will not generate an output

immediately and needs to wait the time advance is elapsed to generate an output.

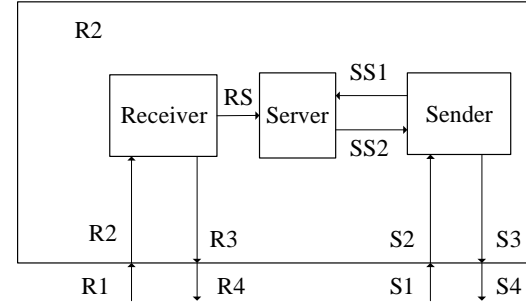


Figure 2.2. Simple couple model of FTP server

The atomic model is a building block for coupled models just as shown in Figure 2.2. The couple model could be built up only by some atomic models or it could contain another couple model as well. The coupled model is a coordinator which distributes the events it receives from the input ports to the correct atomic model or coupled model and send the output it receives from the inside atomic or coupled models to the output ports. Figure 2.3 is a sample showing the definition of the Server shown in Figure 2.2.

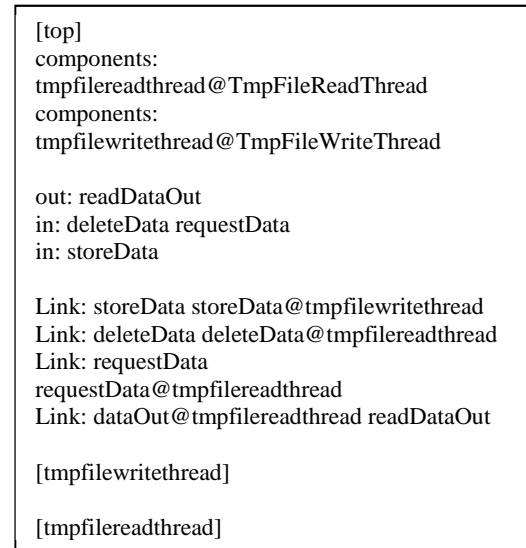


Figure 2.3. Definition of coupled model

2.1.2. Cell-DEVS specification

Cell-DEVS is an extension of DEVS and it supports definition of cellular models. Each cell is an atomic DEVS models and it receives the

input from its neighbor cells and send output to its neighbor cells with same or different time delays. The atomic model of Cell-DEVS has n inputs which is also the number of its neighbors. The inputs are used to activate and compute its local function. The computed result will be sent to the output ports with a time delay. There are two kinds of time delay: transport delay and inertial delay. For transport delay, every local computation result will be output and if there are some other results being output, just put the result into the output queue. For inertial delay, output will be cancelled if a new local computation result is computed and different from that is being output. These two kinds of delay could be used in different situations. For instance, transport delay could be used in registration process as everyone should be registered in the end even at a specific time there might be a lot of people registering simultaneously. Inertial delay could be used in walking process as if there is someone suddenly appearing in front of a person who is walking, the person should stop walking to avoid collision.

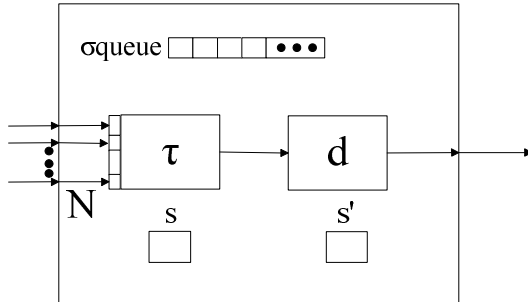


Figure 2.4. Cell-DEVS atomic model [4]

After the behavior of a cell is confirmed and proper delay is chosen, we could build couple models of Cell-DEVS to simulate the group behaviors. There are two sections needed to be paid big attention. One is the neighbors and the other is the rules. The neighbors (shown in figure 2.5) defines the input sources of the cell as the cell only cares about its neighbors' output and take these output as its input to compute its local computing function. The rule section defines the local computing function and the basic format is:

$\{Postcondition\} \text{ Delay } \{Precondition\} \text{ (1)}$

If user defines any state variable inside the Cell-DEVS model, it could also add assignment into the local computing function with the following format:

$\{Postcon.\} \{Assignment\} \text{ Delay } \{Precon.\} \text{ (2)}$

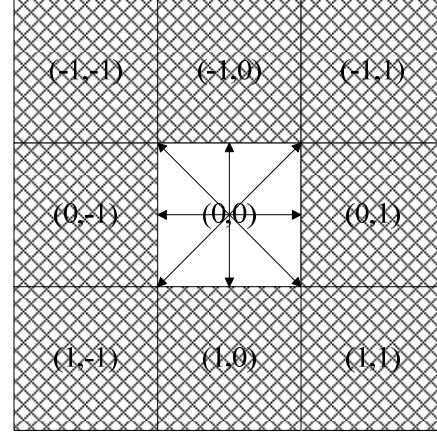


Figure 2.5. Cell and its neighbor

Figure 2.6 shows the basic structure of CD++ Cell-DEVS definition. The rule presented here means: if the cell's neighbor (0,-1) has a non-zero value, the cell has a probability 10% to change its value into that of its neighbor's.

```
[top]
components: crystal
[crystal]
type : cell
width : 50
height : 50
delay : transport
defaultDelayTime : 10
border : nowrapped
neighbors :(-1,-1) (-1,0) (-1,1)
neighbors :(0,-1) (0,0) (0,1)
neighbors :(1,-1) (1,0) (1,1)
initialvalue : 0
initialCellsValue : crystal.val
localtransition : crystal-rule
[crystal-rule]
...
rule:  {(0,-1)}  10  { (0,-1)>0 and
round(uniform(0,10000))> 9000 }
...
```

Figure 2.6. Part of definition of 2D crystal growth

2.2. Crystal growth dynamics and factors

Crystal growth is a typical physical phenomenon. There are a lot of research focusing the dynamics and impacts of different factors [5]. The dynamics of the crystal growth is generally considered as the drop of the boundary energy. Many factors will affect the boundary energy and the most important factors are temperature, impurities and defects. Gravity will have a tiny effect on the growth of the crystal and usually could be ignored. However, in some extreme circumstance where the gravity is very strong, gravity will have a significant impact on the growth of crystal. As follow are a brief introduction of factors that affect the growth of crystal.

2.2.1. Temperature

Temperature has a positive effect on the drop of boundary energy. High temperature will give atom enough energy to relocate itself which means to break the old chemical bond and form new chemical bond. This procedure is analogous as shaking a container filled with balls. After shaking the container several times, you will find the container could be filled with more balls which in the crystal growth case, the boundary energy is dropped. If the shaking force is too weak, the balls will not move and as a result no more space is vacated. This is also the same case in the crystal growth and if the temperature is too low, the crystal will not grow either.

2.2.2. Impurity

Impurities usually have a negative effect on the crystal growth. As mentioned above, to reduce the boundary energy, atom must first break its original chemical bond which means there will be a threshold at the temperature. The impurities will increase this threshold dramatically and in the worst case the atoms will be pinned around the impurities as the chemical bond between them is so powerful. If this situation is spread to the most of the area of the crystal, the crystal will stop growing. This phenomenon is called “dead zone effect”. Thus, in reality the concentration of the impurities must be well controlled and should not exceed a certain level. Nevertheless, the impurities might have a positive effect on the

electronic properties of crystal. Doping techniques such as diffusion and ion planting are widely used to improve the electronic properties of the native crystal. These techniques, however, are beyond the scope of this paper, interested readers could turn to [6] for details.

2.2.3. Defect

The defects are the dislocation of atoms and the simplest defect is vacancy. The defects will not significantly affect the growth of the crystal but will affect the quality of the crystal. The missing atoms will reduce the strength of chemical bond and thus reduce the relocation threshold. A low density of defects would have a positive effect on the growth of crystal but have a negative effect on the lifetime of devices. This is because the heat generated by the circuit on the device is possible to change the structure of the devices which might cause a complete failure of the device. An effective way to reduce defects is annealing and is widely used in industry.

2.3. Overview of this paper

The remainder of this paper is organized as follow: Chapter 3 will introduce the definition of the model, two models will be presented and a detailed implementation will also be presented. Chapter 4 will discuss the simulation results and step by step comparisons based on different factors will also be carried out. Chapter 5 will mainly discuss the conclusion I have got. Chapter 6 will introduce possible improvements for the model in the next version. Chapter 7 will be the acknowledgement.

3. Definition of Model

3.1. Ideal model for crystal growth

Before I present a model that takes temperature, impurities, defects and gravity into consideration, let's first consider an ideal model which does not have impurities or defects. In additional, the gravity is very tiny and could be ignored. Thus, temperature is the only factor that would affect the growth of the crystal. However, there are still a lot of things to do before building a model

which could well describe the reality. In order to simplify the model and also keep the accuracy, two kinds of neighbor atoms are defined: Most adjacent neighbor and Second adjacent neighbor.

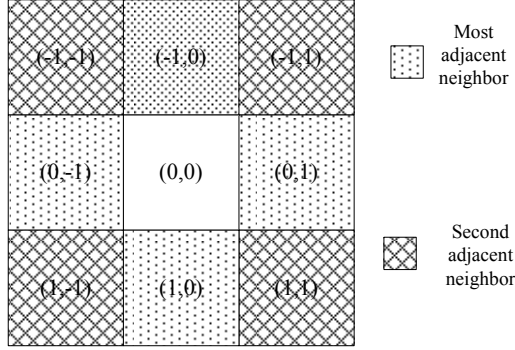


Figure 3.1. Geometry of the cell

A cell will have 9 neighbors including itself. The most adjacent neighbor is directly contacted with the cell while second adjacent neighbor is not directly contacted with the cell but still within the interaction distance. Usually most adjacent neighbor will have a much larger influence than the second adjacent neighbor as in reality the most adjacent neighbors are expected to have chemical bond with the cell. However, as the second adjacent neighbors are still within the interaction distance, their influence could still be significant and sometimes might be larger than the most adjacent neighbors. I will illustrate it later.

3.1.1 Formal specification

The formal specification $\langle X, Y, I, S, \theta, N, d, \tau, \delta_{int}, \delta_{ext}, \lambda, ta \rangle$ for Cell-DEVS crystal is described as follow:

$$X = \{\emptyset\}$$

$$Y = \{\emptyset\}$$

$I = \langle 9, 0, \emptyset, \emptyset \rangle$ // the number of the neighbors is 8, there is no input or output

$S = \{s | s \in [0, 9]\}$ // 0 means defects, 1 to 7 means different crystal orientation, 8 to 9 means impurities (will be used in advanced model).

$$\theta = \{(s, phase, \sigma_{queue}, \sigma), s \in S\}$$

$$phase \in \{active, passive\},$$

$$\sigma_{queue} = \{(v_m, \sigma_m) | m \in N \wedge m \in \infty\} \wedge \forall (i \in N, i \in [1, m]), v_i \in S \wedge \sigma_i \in R_0^+ \cup \infty\},$$

$$N \in \{(-1, -1), (-1, 0), (-1, 1), (0, -1), (0, 0),$$

$$(0, -1), (1, -1), (1, 0), (1, 1)\} \sigma \in R_0^+ \cup \infty$$

$$d = 10 \text{ ms}$$

τ // will be illustrated in 3.1.2

δ_{int} // Done by CD++

δ_{ext} // Done by CD++

λ // Done by CD++

ta (passive) = INFINITY

ta (active) = d

3.1.2 Rules of ideal model

The basic idea of the rules is firstly following the majority principle and then following the probabilistic principle. This is because the cell and its neighbors are forming a group and the majority could have a great influence on the cell. Besides, from the energy point of view, following the majority could further reduce the boundary energy which is dynamic of the growth of crystal. Thus the state of a cell is defined to reflect the decision that is made by the majority within the group. The state of the cell is the current status of cell, and in this simulation, we could roughly consider it as crystal orientation. Although the state of the cells with the same crystal orientation might still be slight different, for instance, one cell might have dangling bond, we could roughly see them have a same state as these tiny differences will usually not affect the transition between different states.

Table 3.1. Crystal state values

State	Color	State values
Empty		0
(100)		1
(010)		2
(001)		3

(110)		4
(101)		5
(011)		6
(111)		7

[top]
components : crystal

[crystal]
type: cell
width: 50
height: 50
delay: transport
defaultDelayTime: 10
border: nowrapped
neighbors: crystal(-1,-1) crystal(-1,0) crystal(-1,1)
neighbors: crystal(0,-1) crystal(0,0) crystal(0,1)
neighbors: crystal(1,-1) crystal(1,0) crystal(1,1)
initialvalue : 0
initialCellsValue : crystal.val
localtransition: cystal-rule

Figure 3.2. Crystal coupled model definition

The model is implemented by CD++ specification language. The size of the cell space is 50×50 and initial cell values are randomly generated through C++ code and written into crystal.val file.

Figure 3.2 shows the definition of cell space, default delay and initial values. The geometry of the model using is shown in figure 3.1 which is also defined in the *neighbors* section. The local computing function is defined in crystal-rule section which will be introduced in figure 3.3.

There are two parts of the rules, first is majority principle which is shown in figure 3.3. The rules are defined as follow:

1. If all the neighbors including the cell itself have a same state, the cell keeps its state unchanged.
2. If any three of the cell's most adjacent neighbors have the same state, the cell will transfer its state to be same as these three neighbors.

3. If none of the above rules are satisfied and any three of its second adjacent neighbors have the same state, the cell will also transfer its state to be the same as these three neighbors.

[cystal-rule]
rule: {(0,0)} 10 {(0,0)>0 and (-1,-1)=(-1,0) and (-1,0)=(-1,1) and (-1,1)=(0,-1) and (0,-1)=(0,0) and (0,0)=(0,1) and (0,1)=(1,-1) and (1,-1)=(1,0) and (1,0)=(1,1)}
rule: {(-1,0)} 10 {(-1,0)>0 and ((-1,0) = (0,-1)) and ((0,-1) = (1,0))}
rule: {(0,-1)} 10 {(0,-1)>0 and ((0,-1) = (1,0)) and ((1,0) = (0,1))}
rule: {(1,0)} 10 {(1,0)>0 and ((1,0) = (0,1)) and ((0,1) = (-1,0))}
rule: {(-1,-1)} 10 {(-1,-1)>0 and ((-1,-1) = (1,-1)) and ((1,-1) = (1,1))}
rule: {(1,-1)} 10 {(1,-1)>0 and ((1,-1) = (1,1)) and ((1,1) = (-1,1))}
rule: {(1,1)} 10 {(1,1)>0 and ((1,1) = (-1,1)) and ((-1,1) = (-1,-1))}
rule: {(-1,1)} 10 {(-1,1)>0 and ((-1,1) = (-1,-1))}

Figure 3.3. Majority principles

Rule 3 reflect the fact if the cell's most adjacent neighbors are not in the same state, their interaction force on the cell might cancel each other. Although in reality this is not always the case, from a rough estimation, this assumption is good enough.

rule: {(-1,-1)} 10 { (-1,-1)>0 and round(uniform(0,10000))> 9000 }
rule: {(0,-1)} 10 { (0,-1)>0 and round(uniform(0,10000))> 8889 }
rule: {(1,-1)} 10 { (1,-1)>0 and round(uniform(0,10000))> 8750 }
rule: {(1,0)} 10 { (1,0)>0 and round(uniform(0,10000))> 8571 }
rule: {(1,1)} 10 { (1,1)>0 and round(uniform(0,10000))> 8333 }
rule: {(0,1)} 10 { (0,1)>0 and round(uniform(0,10000))> 8000 }
rule: {(-1,1)} 10 { (-1,1)>0 and round(uniform(0,10000))> 7500 }
rule: {(-1,0)} 10 { (-1,0)>0 and round(uniform(0,10000))> 6666 }
rule: {(0,0)} 10 { t }

Figure 3.4. Probabilistic principles

The probabilistic principles are shown in figure 3.4. The rules are defined as follow:

1. If any of the majority principles are not satisfied, the cell has a probability to transfer its state to any of its neighbor's state with equal probability. The probability shown in figure 3.4 is 10%.

2. If the cell fails to transfer its state to any of its neighbor's state, it will keep its state not changed

There is an interesting thing in figure 3.4 as the probability that the cell transfers its state to any of its neighbor's should be same. However, the probability in different rules seems to be different. This is because the rules are executed sequentially which means the first rule is always evaluated first. Only if the evaluation of the first rule fails, then the second rule is evaluated. This means if we want to keep the second rule having the same pass probability as the first rule, the probability should be a little bigger than the first rule and the relationship between the expected probability and written probability should satisfy as follow equation:

$$P_{\text{Rule1-fail}} \cdot P_{\text{Rule2}} = P_{\text{expect}} \quad (3)$$

The probability of failure of first rule is 0.9 and the expected probability of second rule is still 0.1, thus the probability of rule 2 is:

$$P_{\text{Rule2}} = \frac{1}{9} \approx 0.1111 \quad (4)$$

This is just the probability used in rule 2 in figure 3.4. Other rule's written probability also follows equation 3.

Figure 3.5 shows the snapshot of the initial state from CD++ modeler which is a part of CD++ toolkit. As mentioned above, the initial state is generated randomly by C++ codes and written to crystal.val. Figure 3.6 shows the C++ codes that generate the initial state.

After 176 iterations, the growth of crystal comes to a stable state. Figure 3.7 shows the stable state and we could see that there are a lot of small single crystals. I will discuss it further in section 4.

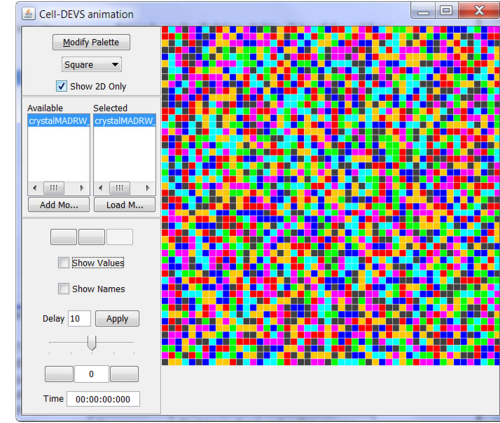


Figure 3.5. Initial state of crystal growth

```
#include <fstream>
#include <stdlib.h>
#include <time.h>
using namespace std;
int main()
{
    srand(static_cast<unsigned>(time(NULL)));
    int m,n;
    ofstream fs;
    fs.open("crystal.val");
    int p=0;
    for(m=0;m<50;m++)
    {
        for(n=0;n<50;n++)
        {
            p= rand()%7+1;
            fs<<"("<<m<<","<<n<<")"<<"="<<p<<endl;
        }
    }
    fs.close();
    return 0;
}
```

Figure 3.6. Initial state generator

3.2 Advanced model for crystal growth

An ideal model of crystal growth is presented and I will take more factors into account. Firstly, let's consider the effect of the defects. To simplify the model, I will only consider the vacancy which is called point defect. In reality, point defect might be filled by its neighbor atom which looks like the point defect is moved. However, this probability is usually relatively small and thus is not taken into consideration in this model. Thus, the point defect will only keep its position and will never transfer its state to any of its neighbor.

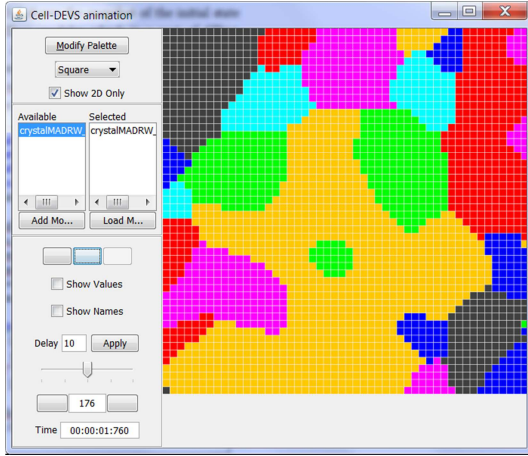


Figure 3.7. Stable State of crystal growth

The effect of impurity is a little complex. There might be several different kinds of impurities. One kind of impurity might be very powerful and thus any atom around it could not change its state and thus its state is pinned. There are also some impurities having a big effect on their neighbors but the effect is not so powerful that the neighbor atoms still have probability to change their states. Although the probability might be much smaller than that of the atoms which do not have any impurities nearby, the state of the atom is still changeable and not pinned.

The gravity will have different effect on the atoms with different states as shown in table 3.2. However, some state as well as its complementary state's crystal orientation is horizontal and thus, the affection of gravity will be same on them. The final impact of gravity is determined by the difference between different states, for instance, the correction value of gravity that state (001) transfer to state (100) is 0.02 while for state (100) to state (001) is -0.02. This reflects the fact that if it is easier to transfer one state to another state, it must be harder for the reverse process.

Table 3.2 State value and gravity additional value

State	Color	State value	Gravity
Empty		0	0
(100)		1	0.01

(010)		2	0.01
(110)		3	0.01
(111)		4	0.02
(101)		5	0.02
(011)		6	0.02
(001)		7	0.03
Impurity A		8	0
Impurity B		9	0

Thus the correction value of the probability is described as follow:

$$P_{actual} = P_{original} + P_{impurity} + P_{gravity} \quad (5)$$

The probability of $P_{impurity}$ is the correction value of impact from the impurities. As mentioned above, two types of impurities are taken into consideration. Type A impurity with the state value of 8 will pin all the atoms around it while type B impurity with the state value of 9 will have a 50% probability to allow its neighbors to change its state. However, if there are two or more impurities with type B within the interaction distance of a specific atom, the atom will also be pinned and could not change its state. This indicates heavy impurities will stop the crystal from growing even the effect of the impurities might not be very strong.

As the effect of impurities is involved in the model, another issue is also involved as how to keep a same probability that the atom transfer its state to its neighbor. In the previous model, there is no impurity or defects and thus the second rule is evaluated only if the probability part of its previous rule fails. However, in the new model, the evaluation might still fail as it is point defect or impurity, thus, to keep the equality of probability, the number of the impurity and defects should be taken into consideration when calculating the probability.

```
rule : { (0,-1) } 10 { (0,-1)>0 and (0,-1)!=9 and
( ( (-1,-1)=9 and ( round(uniform( 0,1000000 ) )
< ( 120000+400*( (0,-1)+(0,0)*(-1) ) ) ) ) or ( (-
1,-1)!=9 and ( round( uniform( 0,1000000 ) ) <
( 136301+454*( (0,-1)+(0,0)*(-1) ) ) ) ) }
```

Figure 3.8 Sample rule for advanced crystal growth model

Figure 3.8 shows the sample rule for the advanced crystal growth model. In this rule, two factors are taken into consideration, impurity and gravity. If cell (-1,-1) is impurity, then the probability should start with 0.12 plus the correction value of the gravity but not 0.136 plus the correction value of the gravity. This is because previous rule does not have a failure rate of 88% but 100% if (-1,-1) is impurity. There is interesting thing here as if there is a point defect, the probability is increased. As mentioned in part 2, low density of point defect will slightly increase the probability. This indicates the cell has a larger probability to transfer its state to the cell which is around the point defect. Thus, the rule is just as expected.

```
[top]
components : crystal

[crystal]
type: cell
width: 50
height: 50
delay: transport
defaultDelayTime : 10
border: nowrapped
neighbors: crystal(-1,-1) crystal(-1,0) crystal(-1,1)
neighbors: crystal(0,-1) crystal(0,0) crystal(0,1)
neighbors: crystal(1,-1) crystal(1,0) crystal(1,1)
initialvalue : 0
initialCellsValue : crystal.val
neighborports: pin
localtransition : crystal-rule
```

Figure 3.9 Definition of advanced crystal growth model

Figure 3.9, 3.10 and 3.11 shows the full definition and rules for the advanced crystal growth model. Comparing with the idea model, there are several changes both in the majority principles and the probabilistic principles. The changes of the majority parts are described as follow:

1. If current cell is vacancy or impurity, the cell will keep the state unchanged.

```
[crystal-rule]
rule : {(0,0)} 10 { (0,0)=9 or (0,0)=0 or
( statecount(8,~pin)>0 or statecount(9,~pin)>1 )
or ( statecount(9,~pin) =1 and
round(uniform( 0,1000000 ) ) > 500000 ) }
rule : {(0,0)} 10 {(0,0)>0 and (-1,-1)=(-1,0) and
(-1,0)=(-1,1) and (-1,1)=(0,-1) and (0,-1)=(0,0)
and (0,0)=(0,1) and (0,1)=(1,-1) and (1,-1)=(1,0)
and (1,0)=(1,1)}
rule : {(-1,0)} 10 {(-1,0)>0 and (-1,0)!=9 and ((-1,0) = (0,-1)) and ((0,-1) = (1,0))}
rule : {(0,-1)} 10 {(0,-1)>0 and (0,-1)!=9 and
((0,-1) = (1,0)) and ((1,0) = (0,1))}
rule : {(1,0)} 10 {(1,0)>0 and (1,0)!=9 and ((1,0) = (0,1)) and ((0,1) = (-1,0))}
rule : {(0,1)} 10 {(0,1)>0 and (0,1)!=9 and ((0,1) = (-1,0)) and ((-1,0) = (0,-1))}
rule : {(-1,-1)} 10 {(-1,-1)>0 and (-1,-1)!=9 and
((-1,-1) = (1,-1)) and ((1,-1) = (1,1))}
rule : {(1,-1)} 10 {(1,-1)>0 and (-1,-1)!=9 and
((1,-1) = (1,1)) and ((1,1) = (-1,1))}
rule : {(1,1)} 10 {(1,1)>0 and (1,1)!=9 and ((1,1) = (-1,1)) and ((-1,1) = (-1,-1))}
rule : {(-1,1)} 10 {(-1,1)>0 and (-1,1)!=9 and ((-1,1) = (-1,-1))}
```

Figure 3.10 Majority principles

2. If there is type A impurity or more than one type B impurity within the interaction distance of a specific cell, the cell will also keep its state unchanged and the state is pinned.
3. If there is only one type B impurity around the cell, the cell has 50% probability to change its state. However, it still needs to follow other rules presented in the ideal model.
4. If there is no impurity around the cell, and for the majority principles parts, the cell is still following the rule same as ideal model.

For the probabilistic principles parts, the changes are as follow:

1. The probability is combined with the correction value of gravity, which means the probability that transfers to different state will be slightly different.
2. If there is one type B impurity around the cell, the probability that the cell changes its state to the state of its neighbor is still equal (if not considering the gravity). In the case presented in Figure 3.11, the cell has a probability (roughly 0.12) to change its state to that of its neighbor.

```

rule : {( -1,-1)} 10 { ( -1,-1)>0 and ( -1,-1)!=9
and ( round(uniform( 0,1000000 ) ) <
( 120000+400*( ( -1,-1)+(0,0)*(-1) ) ) ) }
rule : {(0,-1)} 10 { (0,-1)>0 and (0,-1)!=9 and
( ( ( -1,-1)=9 and
( round(uniform( 0,1000000 ) ) <
( 120000+400*( (0,-1)+(0,0)*(-1) ) ) ) ) or ( ( -1,-1)!=9 and
( round( uniform( 0,1000000 ) ) <
( 136301+454*( (0,-1)+(0,0)*(-1) ) ) ) ) ) }
rule : {(1,-1)} 10 { (1,-1)>0 and (1,-1)!=9 and
( ( ( ( -1,-1)=9 or (0,-1)=9 ) and
( round(uniform( 0,1000000 ) ) <
( 136301+454*( (1,-1)+(0,0)*(-1) ) ) ) ) or ( ( ( -1,-1)!=9 and
(0,-1)!=9 ) and
( round( uniform( 0,1000000 ) ) <
( 157900+526*( (1,-1)+(0,0)*(-1) ) ) ) ) ) }
rule : {(1,0)} 10 { (1,0)>0 and (1,0)!=9 and
( ( ( ( -1,-1)=9 or (0,-1)=9 or (1,-1)=9 ) and
( round(uniform( 0,1000000 ) ) <
( 136301+454*( (1,0)+(0,0)*(-1) ) ) ) ) or ( ( ( -1,-1)!=9 and
(0,-1)!=9 and (1,-1)=9 ) and
( round( uniform( 0,1000000 ) ) <
( 187500+625*( (1,0)+(0,0)*(-1) ) ) ) ) ) }
rule : {(1,1)} 10 { (1,1)>0 and (1,1)!=9 and
( ( ( ( -1,-1)=9 or (0,-1)=9 or (1,-1)=9 or
(1,0)=9 ) and ( round(uniform( 0,1000000 ) ) <
( 187500+625*( (1,1)+(0,0)*(-1) ) ) ) ) or ( ( ( -1,-1)!=9 and
(0,-1)!=9 and (1,-1)=9 and
(1,0)!=9 ) and ( round( uniform( 0,1000000 ) ) <
( 230767+769*( (1,1)+(0,0)*(-1) ) ) ) ) ) }
rule : {(0,1)} 10 { (0,1)>0 and (0,1)!=9 and
( ( ( ( -1,-1)=9 or (0,-1)=9 or (1,-1)=9 or (1,0)=9
or (1,1) = 9 ) and
( round(uniform( 0,1000000 ) ) <
( 230767+769*( (0,1)+(0,0)*(-1) ) ) ) ) or ( ( ( -1,-1)!=9 and
(0,-1)!=9 and (1,-1)=9 and
(1,0)=9 ) and ( round( uniform( 0,1000000 ) ) <
( 300000+1000*( (0,1)+(0,0)*(-1) ) ) ) ) ) }
rule : {( -1,1)} 10 { ( -1,1)>0 and ( -1,1)!=9 and
( ( ( ( -1,-1)=9 or (0,-1)=9 or (1,-1)=9 or (1,0)=9
or (1,1) = 9 or (0,1)=9 ) and
( round(uniform( 0,1000000 ) ) <
( 300000+1000*( ( -1,1)+(0,0)*(-1) ) ) ) ) or ( ( ( -1,-1)!=9 and
(0,-1)!=9 and (1,-1)=9 and (1,0)=9 and
(1,1)!=9 and (0,1)=9 ) and
( round(uniform( 0,1000000 ) ) <
( 428600+1429*( ( -1,1)+(0,0)*(-1) ) ) ) ) ) }
rule : {( -1,0)} 10 { ( -1,0)>0 and ( -1,0)!=9 and
( ( ( ( -1,-1)=9 or (0,-1)=9 or (1,-1)=9 or (1,0)=9
or (1,1) = 9 or (0,1)=9 or ( -1,1)=9 ) and
( round(uniform( 0,1000000 ) ) <
( 428600+1429*( ( -1,0)+(0,0)*(-1) ) ) ) ) or ( ( ( -1,-1)!=9 and
(0,-1)!=9 and (1,-1)=9 and (1,0)=9 and
(1,1)!=9 and (0,1)!=9 and ( -1,1)!=9 ) and
( round( uniform( 0,1000000 ) ) <
( 750000+2500*( ( -1,0)+(0,0)*(-1) ) ) ) ) ) }
rule : {(0,0)} 10 { t }

```

Figure 3.11 Probabilistic principles

However, as mentioned in the majority principles, the cell around type B only has 50% probability to change its state. Thus, the actual probability is $0.12/2=0.06$, which is much lower than the cells without the affection of impurities.

4. Simulation results and discussion

After finishing the implementation of the ideal model and advanced model, I will discuss the simulation results. I will compare the impact of different factors on the growth of crystal. The ideal model is very convenient to study the impact of temperature while advanced model is desired to study the other factors. In the later section, I will discuss them individually.

4.1 Temperature

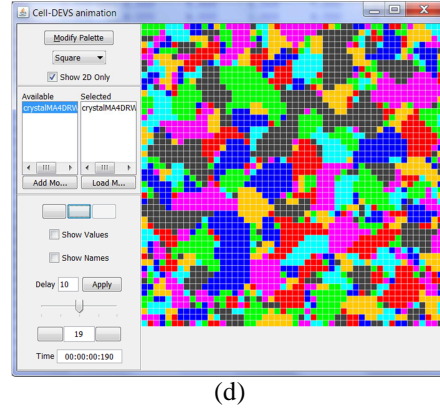
Temperature is key factor for the growth of crystal. It is expected to gain a bigger crystal with higher crystal. As the temperature goes up, the cell will gain enough energy and thus is more easily to change its state. This indicates as the temperature goes up, the probability that a cell changes its state will also grow. This corresponds with the probabilistic principles parts of the implementation shown in figure 3.4. In order to study the impact of different temperature, ideal model instead of advanced model is used to keep proper isolation. In additional, different probabilities will also be compared. Table 4.1 shows the probabilities that will be tested.

Table 4.1 Different probabilities being tested

State count	Probability
7	0.12
7	0.1
7	0.05
7	0.02

In order to make a proper comparison, the initial state of these four different probabilities is kept same and the difference is only the probability. Figure 4.1 (a) ~ (d) shows the simulation results based on different probabilities. As it could be expected, the average size of crystal with

probability 0.12 is the largest as the corresponding temperature is the highest. Besides, the shape of the crystal is becoming much more symmetry as the probability goes up. This alludes that if we want to gain a symmetry crystal, enough temperature must be provided. The growth of figure 4.1 (d) is stopped very soon after the beginning. This indicates the growth of crystal needs enough temperature. If the temperature is below a certain level, the growth crystal will not begin.

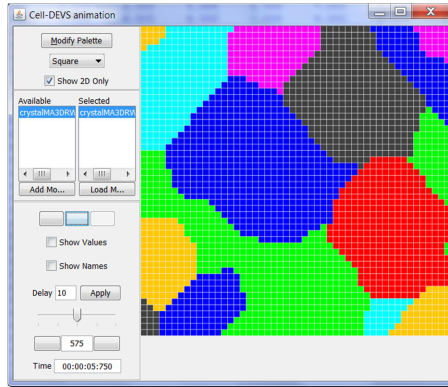


(d)

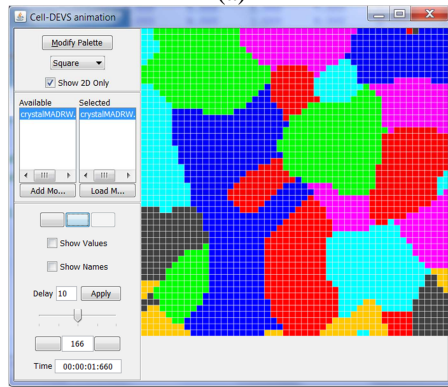
Figure 4.1 Simulation results of different probabilities (a) 0.12 (b) 0.1 (c) 0.05 (d) 0.02

Another thing that we could see from the simulation result is the boundary pin effect as shown in figure 4.2. Although the total simulation takes about 575 steps, the shape of the crystals are almost fixed in step 180. This is because the boundary will have a pin effect on the shape of the crystal. Before the boundary is formed, the cell could easily change its state as it wants. However, after boundary is formed, the cell's state is usually pinned and almost keeps unchanged until the end of the process.

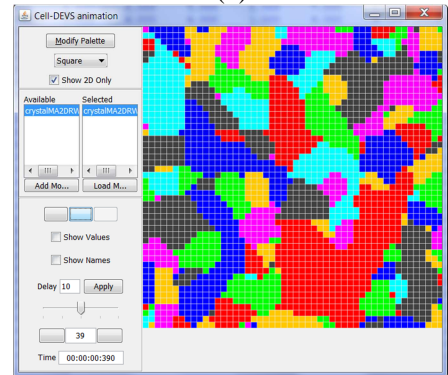
Boundary pin effect is common phenomenon in the crystal growth. The real situation might be a little complicated than the case shown in the simulation result. This is because along the boundaries, there might be a lot of defects, impurities. However, the basic principle is almost the same. I will illustrate the boundary pin effect further from mathematical and physical view of point individually.



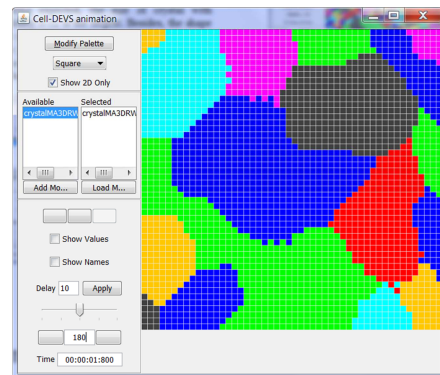
(a)



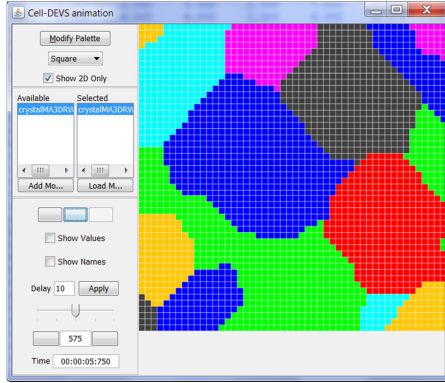
(b)



(c)



(a)



(b)

Figure 4.2. Step 180 (a) and step 575 (b)

Figure 4.3 shows the boundary between two crystals. Five neighbors of the central cell belong to one single crystal and the other three belong to another. The cell in the middle has a probability of 0.12 to transfer its state to its neighbor's state, however, as 5 of its neighbors has the same state with itself, thus the cell will only have probability of $0.12 \times \frac{3}{8} = 0.045$ to change its state. This is a very low probability indeed. The mathematic deduction indicates the boundary cells will have a much lower probability to change its state and as a result the shape of the crystal is pinned. From physical point of view, there is usually a large tensile force around the boundary which makes the boundary cell hard to change its state. The boundary tensile force will compensate the energy that the cell gains from heat moments. Thus the residue energy of the cell is very weak and could not change its state. To break the boundary restriction, a much higher temperature needs to be provided. This is also the result shown in figure 4.1.

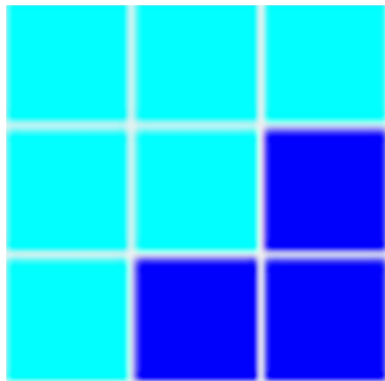
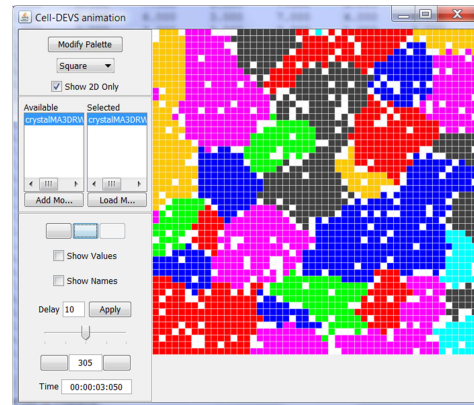


Figure 4.3. Crystal boundary

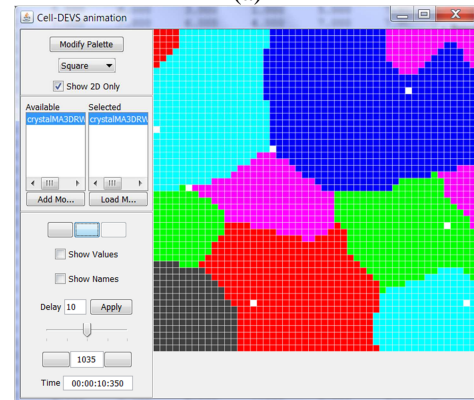
4.2. Defects

A small number of point defects will have a slightly effect on the crystal growth. However, large number of point defects will have a big effect on the growth of crystal. This is because the interaction between atoms is weakened and thus the atoms could not interact with each other to form a bigger crystal.

Figure 4.4 shows the impact of defects. If the density of the defects is relatively small, the growth of crystal is not affected and similar to the ideal case, while under large density of the defects, the crystal is much smaller. This is because the defects around the boundary of the crystals will weak the interaction between different crystals and as a result the boundary atoms will have a much lower probability to change its state. This will in return limit the growth of crystal. The defects will also affect the quality of the crystal as the chemical bond between different atoms is hereby weakened and thus is very easy to change its location. This would reduce the lifetime of device base on it.



(a)



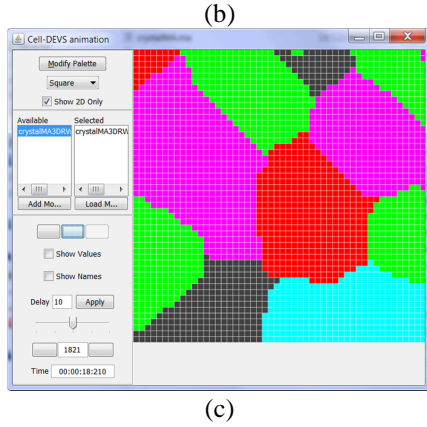


Figure 4.4. Different densities of defects

4.3. Impurity

The impurity will have a negative effect on the growth of crystal even with a small concentration. Different from defects, the impurities will have a strong interaction force with its nearby atoms. The chemical bond between the impurities and the atom of the crystal is usually much more secure than the chemical bond between the atoms of the crystal. This alludes that the state of this kind of chemical bond is much more stable. Break this state usually need much higher temperature. However, some impurities are very powerful and thus even the crystal has changed its physical objects, the bond is still not easy to break. There are also some impurities whose chemical bond could be easily broken. In this simulation, we use state 8 to represent the first kind of impurities and state 9 to represent the second kind of impurities. In order to better illustrate the impact of impurities, defects as well as gravity are not taken into consideration.

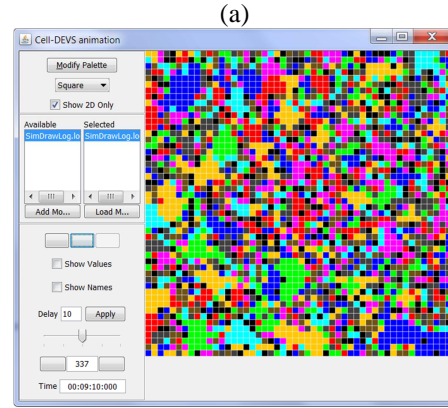
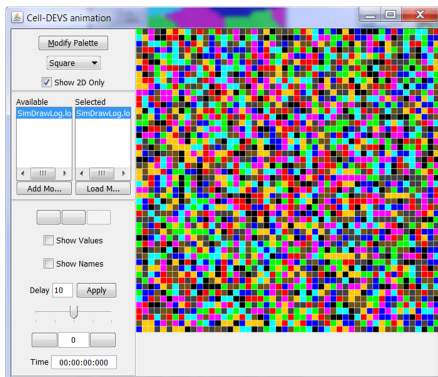


Figure 4.5. Heavy impurities (a) initial state (b) final state

From figure 4.5, we could see that under heavy impurities condition, the crystal almost stop growing.

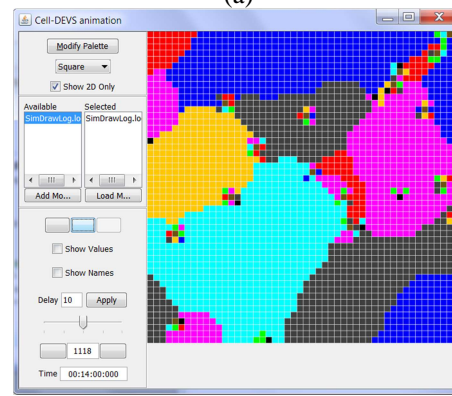
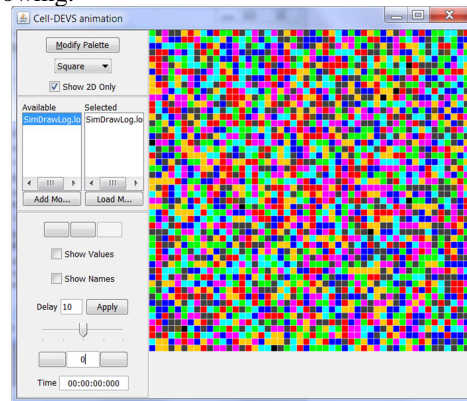


Figure 4.6 Light impurities (a) initial state (b) final state

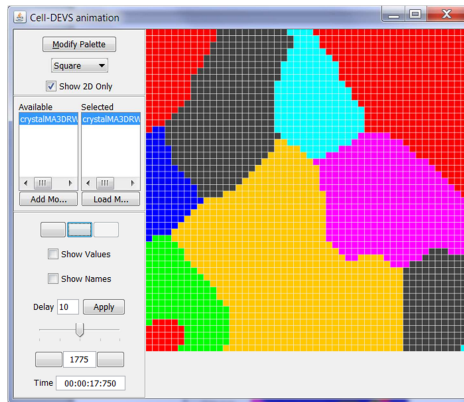
Figure 4.6 shows a case with light impurities. There are some groups of impurities inside the

crystal or along the boundary of different crystals. However, the impurities do not significantly affect the growth of the crystal. This indicates if we keep the concentration of impurities in the relatively low value, large size of crystal is still possible.

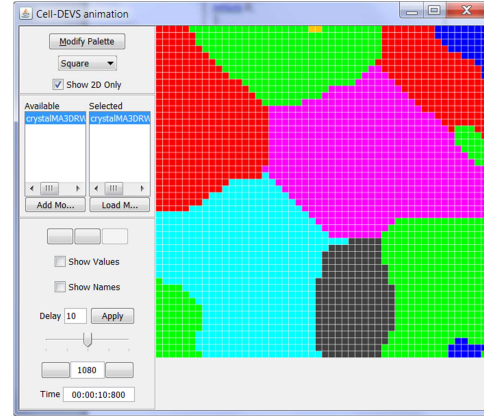
4.4. Gravity

Although the atom is very tiny, it still has a mass and thus gravity will have impact on it. As a result, gravity will indirectly affect the growth of crystal. The probability that one state of the atom transfers to another will slightly increase if the gravity potential drops. Figure 4.7 shows the impact of gravity. The correction value of figure 4.7 (b) is defined in table 3.2 while the value of figure 4.7 (c) is ten times larger than that of figure 4.7 (b). From the figure, we could see that light gravity does not significantly affect the growth of the crystal. As from figure 4.7 (a) and (b), we could not see a particular state have a dominated priority.

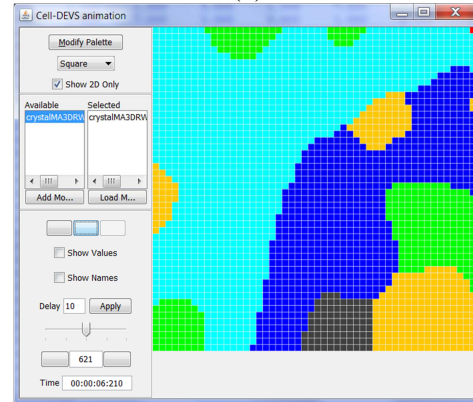
The case is a little different in figure 4.7 (c), the (111) state has a dominated priority and almost half of the cell space is filled by this state. This phenomenon accords with our understanding of gravity. Only if the gravity is very strong, the impact of the gravity will be seen. In the normal case, especially in our earth environment, the gravity has a slightly impact on the growth of crystal and usually could be ignored. However, gravity will affect the symmetry of the crystal and thus perfect crystal is not very easy to be got in gravitational field.



(a)



(b)



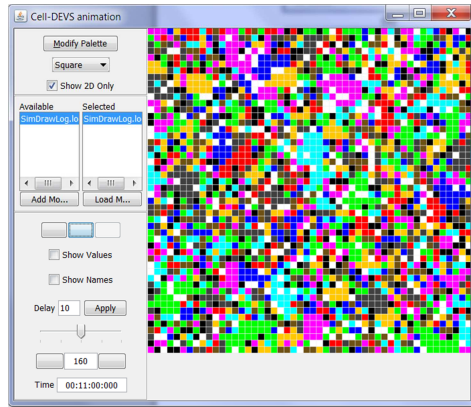
(c)

Figure 4.7 Impact of gravity (a) no gravity (b) light gravity (c) large gravity.

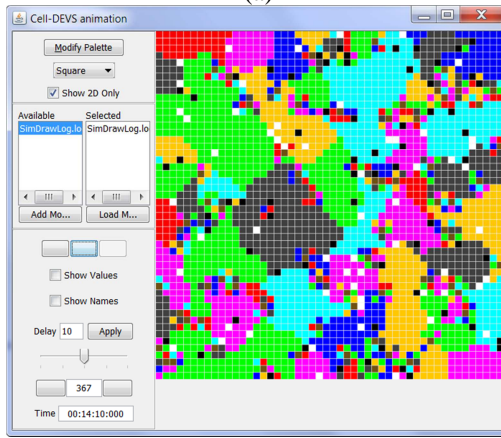
4.5. Combination impacts of the factors

We have discussed the impact of different factors individually in the previous sections. Now we will combine these factors and see how they affect the growth of crystal together.

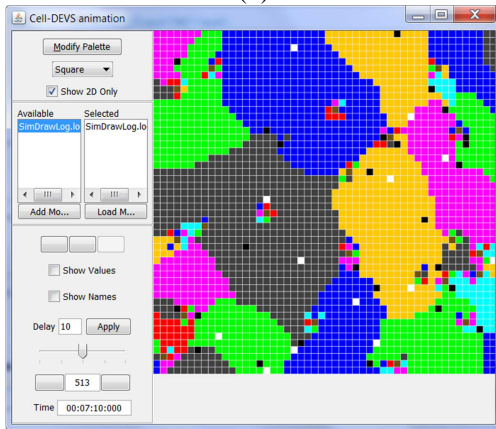
Figure 4.8 shows the combined effects of the temperature, impurities, defects and gravity. We could see that as the concentration of the impurities and defects goes up, the impact of temperature goes weaker. In the heavy impurities and defects situation, the impact of temperature could be ignored as almost all the atoms are pinned and could not change their states. From the macroscopic angle, the crystal has stopped growing.



(a)



(b)



(c)

Figure 4.8 Combination impacts (a) heavy impurities and defects (b) medium impurities and defects (c) light impurities and defects

5. Conclusion

I have investigated the impact of temperature, impurity, defect and gravity on the growth of crystal. Details of the impact from different factors are well studied. The simulation shows

some valuable results and they are summarized as follow:

1. As the temperature goes up, the growth of crystal becomes more active. It is easier to gain bigger crystal with higher temperature.
2. The boundary atoms have a much lower probability to change its state and thus the boundary will have a pin effect on the shape of the crystal.
3. The impurities will also have pin effect on the atoms, heavy impurities will cause the crystal to stop growing while light impurities does not have a significant effect on the growth of the crystal. This means the concentration of the impurity should be kept in a low level.
4. The defects do not have a significant effect on the growth of the crystal. However, large density of defects will weaken the interaction force between the atoms and thus limit the size of the crystal.
5. Gravity will have a tiny effect on the growth of crystal, especially in the earth gravitational field. However, as simulation result shows, strong gravity could have a strong impact on the growth of crystal. This case is very possible to happen in some planet with a large mass.

In total, the impurity and temperature are the most important factors during the growth of crystal and these two factors will determine the size of the crystal. The other two factors have a slight impact on the growth comparing with the impurity and temperature. In additional, in the light impurities situation, we could roughly consider the impact of temperature only. Nowadays, the concentration of the impurity is usually very low, which is several magnitudes lower than the atoms. This indicates in the most case, the ideal model could work very well. This could help us to reduce the computation as well as the model complexity.

6. Future work

I have investigated the impacts of four factors and get some valuable results. However, there are still some potential improvements could be done in the future work.

1. Only point defect is considered in the model. Some other defects such as line defects, planar defects and etc. are not considered in the model. Besides, the defects could not move which does not accord with the reality.

2. Rectangular geometry is used in the model. However, this does not reflect the topology of the reality. Instead, hexagonal geometry should be used to gain more realistic results.

3. 2D crystal growth is also affected by the atoms from other layers. This indicates 3D simulation will reflect a more benign reality.

In the next version of model, these three factors could be taken into account.

7. Acknowledge

I appreciate a lot for the supervision and support from Prof. Wainer.

8. Reference

[1] J. Ameghino, A. Troccoli, and G. Wainer, "Models of Complex Physical Systems Using Cell-DEVS", *Proceedings of 34th Annual Simulation Symposium*, 2001, pp. 266-273.

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

[3] E. Gamma, R. Helm, R. Johnson, and J. Vlissides, *Design Patterns: Elements of Reusable Object-Oriented Software*, Addison-Wesley, USA, 1994

[4] R. Madhoun, and G. Wainer "Modeling Space-Shaped Defense Applications with Cell-DEVS", *Proceedings of SISO Fall Interoperability Workshop*, 2005

[5] Y. He, H. Ding, L. Liu, and K. Shin, "Computer simulation of 2D grain growth using a cellular automata model based on the lowest energy principle", *Materials Science and Engineering*, vol.429, iss. 1-2, August 2006, pp. 236-246.

[6] C. Hamaguchi, *Basic semiconductor physics*, sprinter-verlag, German, 2001.