Biophysics
A Computational Approach
Concepts, Models, Methods and Algorithms
Lecture 4: Growth, Aggregation and Deposition

Dieter W. Heermann

Heidelberg University

November 21, 2016

Table of Contents

Introduction I

In essence many of the phenomena associated with growth, aggregation and deposition can be thought of in terms of particles diffusion problems [1]. A particle diffuses through a medium until it gets in contact with either another particle of a cluster of particles. Depending on the model the particles sticks or is reflected a number of times. This simple model in its variations able to reproduce many of the structures one observes.
On the other hand there are cellular automata-based models.
Assume an object with N elements at positions r_i with unit mass. We define the **radius of gyration** of the object by

$$R_g^2 = \frac{1}{2N^2} \sum_{i,j=1..N} (r_i - r_j)^2 \quad . \tag{1}$$

For later purposes we note that an alternative to this approach is to define this radius via the principle moments of the **gyration tensor** S

$$S_{mn} = \frac{1}{N} \sum_{i=1}^{N} r_i^{(m)} r_i^{(n)} \tag{2}$$

where we assume

$$\sum_{i=1}^{N} r_i = 0 \tag{3}$$

$$R_g^2 = \lambda_1^2 + ... + \lambda_d^2 \quad . \tag{4}$$

Another possibility is to define the extend of the object by the smallest box that is needed such that object fits into the box

$$R_b = \max_{i,j=1,\ldots,N} |r_i - r_j| \quad . \tag{5}$$

We further define the **asphericity**

$$b = \lambda_d^2 - \frac{1}{d-1}(\lambda_1^2 + ... + \lambda_{d-1}^2) = \frac{d}{d-1}\lambda_d^2 - \frac{R_g^2}{2} \tag{6}$$

Let R be the radius of the cluster (R_g, R_b) and M be the mass (here the number of occupied lattice sites N) that belongs to cluster, then

$$M = N \propto R^{d_f} \tag{7}$$

describes the relationship between the radius and the mass where we anticipate that the object may not be compact but be fractal with the fractal dimension $d_f < d$.
Let n_s be the number of sites at the surface and h_i the distance from a reference distance measuring the height of the surface. Then average height is given by

$$\langle h \rangle = \frac{1}{n_s} \sum_i h_i \tag{8}$$

Introduction III

From this we derive *surface roughness*

$$\sigma^2 = \frac{1}{n_s} \sum_i (h_i - \langle h \rangle)^2 \tag{9}$$

Introduction IV

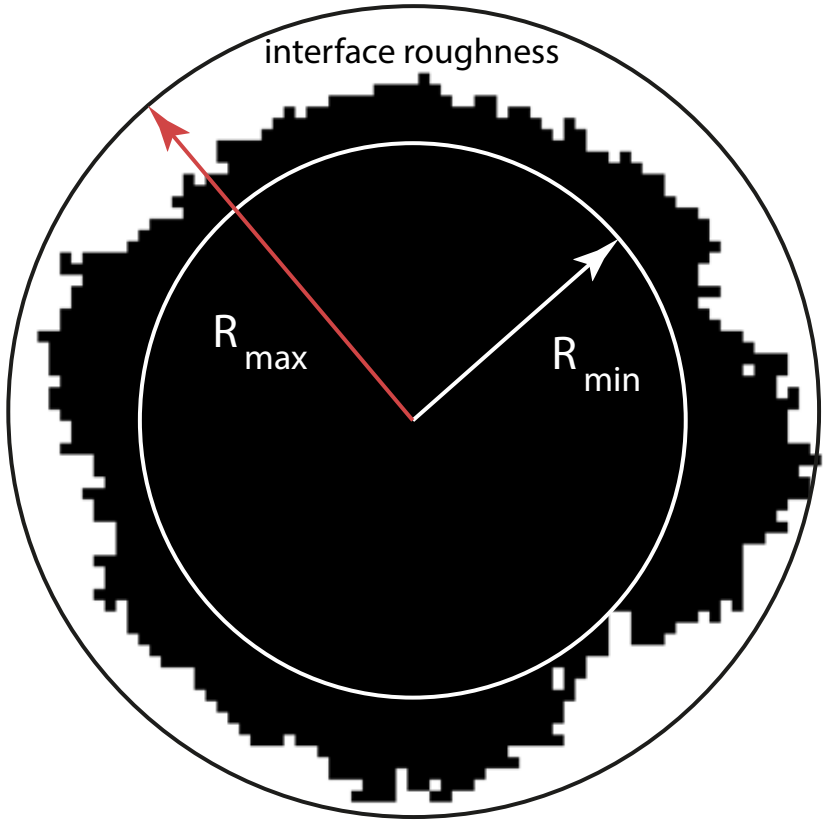

Figure: Interface roughness

Particle Systems I

We shall start with a lattice like $\Lambda = L^d$ or $\Lambda = \mathbf{Z}_n \times \mathbf{Z}_m$, where n and m are integers, or a graph $\Lambda = G$. We will call a system a **particle systems** if

1. each site $s \in \Lambda$ is in one of a finite number q of states, and
2. each site can change its state depending on the number of neighbouring sites

The time evolution of the particle system is described by a Discrete Time Markov Chain. Let $s(t)$ be the state of the site s at time t Then the particle system changes its state $\Lambda(t) \to \Lambda(t+1)$ by the rate $q(s, s')$, where s' denotes one of the possible finite state that s can be in.

Updating rules

- **synchronous**: synchronous updating of a discrete time process which updates all of the sites simultaneously
- **asynchronous**: a site is chosen at random

Growth Models: Eden Cluster I

Eden[2] introduced a stochastic growth model which may be used to study the proliferation of bacteria in a culture medium, propagation of epidemics, chemical reactions, tumor growth etc. In the simplest variant of the model every site on the periphery of the object has an equal probability of being selected as the next growth site.
For simplicity, we assume that the growth takes place on a lattice $\Lambda = L^d$. Once a lattice site s is initially chosen to be the seed, then the nearest neighbour sites are the possible growth sites. Each of the perimeter sites is visited and given the chance to change its state to being occupied. Once a site is occupied the nearest neighbour sites that are not already part of the cluster are added to the list of perimeter sites. This procedure is iterated.

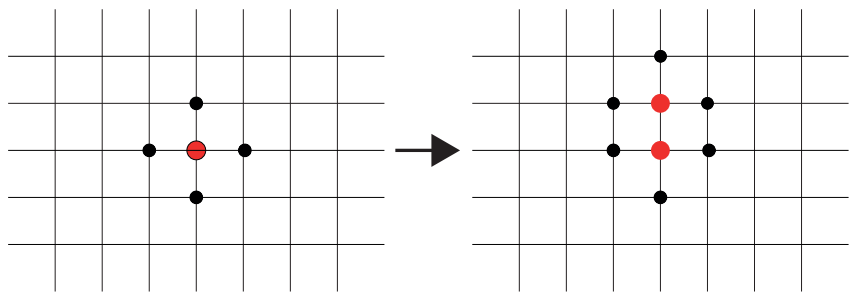

Figure: Steps in the Eden cluster growth. The left panel shows and initial occupied (red) site. A site of the perimeter (black) is chosen during the next step and given the chance to change to being occupied. The right panel shows the situation after the perimeter site has changed its state and additional perimeter sites have been added.

Growth Models: Eden Cluster II

UNIVERSITÄT HEIDELBERG ZUKUNFT SEIT 1386

Algorithm 1 Basic Algorithm: Growth

choose initial site s
add first site to *perimeter_list*
for *n_cycles* **do**
  len = length of *perimenter_list*
  for *len* **do**
    i iid from $\{0, .., len - 1\}$
    s = select at random one of the nearest neighbours of *perimeter_list*(i)
    if s not in *perimeter_list* and not in *new_sites_list* **then**
      add s to *new_sites_list*
    end if
  end for
  add *new_sites_list* to *perimeter_list*
end for

Growth Models: Eden Cluster

UNIVERSITÄT HEIDELBERG ZUKUNFT SEIT 1386

```
import random
mcs_max = 100
random.seed(4711)

s = (0,0)
perimeter_list = []
new_sites_list = []
perimeter_list.append(s)

for mcs in range(mcs_max):
  for n in range(len(perimeter_list)):
    i = random.randrange(0,len(perimeter_list), 1)
    s = perimeter_list[i]
    d = random.randrange(0,4,1)
    if d == 0:
      sn = (s[0],s[1]-1)
    elif d == 1:
      sn = (s[0],s[1]+1)
    elif d == 2:
      sn = (s[0]-1,s[1])
    elif d == 3:
      sn = (s[0]+1,s[1])
    elif d == 4:
      print "should not happen"
    if sn not in perimeter_list:
      new_sites_list.append(sn)
  perimeter_list.extend(new_sites_list)
  new_sites_list = []
```

Code 1: eden.py

Growth Models: Eden Cluster

One starts from one occupied site on a lattice as a seed for a cluster. At every “time step" one additional randomly selected perimeter site is occupied. A perimeter site is an empty neighbour of an already occupied site.

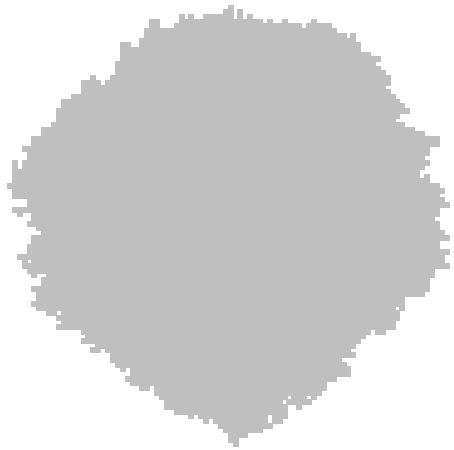

Figure: Eden-cluster

- The Eden cluster is anisotropic due to the underlying lattice, i.e. its shape tends to lengthen along the directions of the lattice axes.
- Initially unoccupied sites in the interior eventually get filled in
- There is no scale invariance
- the cluster is self-affine

Voter Model

Let $\Lambda = L^d$ be a lattice. Let f be a function (usually increasing). For each site s of the lattice a set of neigbours (nearest, next-nearest, etc.) is chosen. Start with a seed which is set to be occupied. Choose at random a site of the lattice. Let *count(s)* be the number of occupied neighbour sites of s. The site s changes its state to occuied with probability $f(count(s))$.
A variation on this that an occupied site becomes occupied at a rate $delta/(1 + count(s))$. A unoccupied site becomes occupied at a rate equal to $1/(1 + count(s))$

Leath Cluster Model

UNIVERSITÄT HEIDELBERG ZUKUNFT SEIT 1386

Algorithm 2 Basic Algorithm: Leath

choose initial site s
add s to visited list
add neighbours of s to *perimeter_list*
while perimeter_list not empty and *max_sites* not reached **do**
  select site s and delete from *perimenter_list*
  add s to visited sites
  if $p <$ random number **then**
    add s to *sites_list*
    add neighbours of s to *perimeter_list*
  end if
end while

Leath Cluster Model

Implementation using a **stack**. Here the *perimenter_list* is a stack where the **pop** operation deletes the element from the stack. Since the **push (append)** put the next perimeter site right at the top of the stack, the growth may proceed 'in perferred direction' as the example in Figure 4 shows. the growth here terminated due to the criterion of 'maximum number of sites reached condition. While in principle the implementation is correct if beside the natural termination criterion no other is used, in practice the implementation is not correct. A *deque* implementation corrects the situation.

```
def addSiteToList(s,sites_list):
    sites_list.append(s)
    return sites_list

def addNeigboursToList(s,perimeter_list,sites_visited_list):
    sn = (s[0],s[1]-1)
    if (sn not in perimeter_list and sn not in sites_visited_list):
        perimeter_list.append(sn)
    sn = (s[0],s[1]+1)
    if (sn not in perimeter_list and sn not in sites_visited_list):
        perimeter_list.append(sn)
    sn = (s[0]-1,s[1])
    if (sn not in perimeter_list and sn not in sites_visited_list):
        perimeter_list.append(sn)
    sn = (s[0]+1,s[1])
    if (sn not in perimeter_list and sn not in sites_visited_list):
        perimeter_list.append(sn)
    return perimeter_list
```

Code 2: Leath cluster growth algorithm part 1: functions

Leath Cluster Model


```
maxSites = 100000
random.seed(4711)

# Initialize
p  = 0.5
s  = (0,0)
perimeter_list      = []
sites_list          = []
sites_visited_list  = []
addSiteToList(s,sites_list)
addSiteToList(s,sites_visited_list)
addNeigboursToList(s,perimeter_list,sites_list)

# Main loop
while (len(perimeter_list) > 0 and len(sites_list) < maxSites):

    s = perimeter_list.pop()
    addSiteToList(s,sites_visited_list)
    if (random.random() < p):
       addSiteToList(s,sites_list)
       addNeigboursToList(s,perimeter_list,sites_list)
```

Code 3: Leath cluster growth algorithm part 2

Leath Cluster Model

Figure: **stack**

Leath Cluster Model

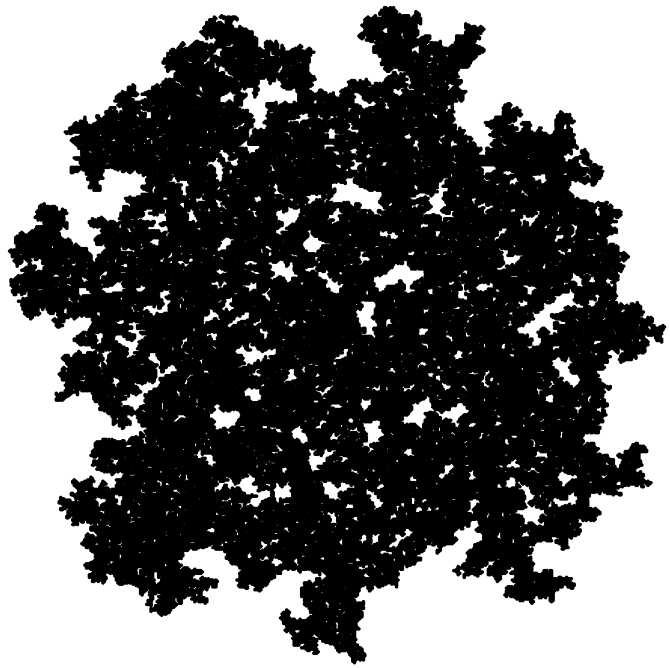

Figure: Leath cluster at $p = 0.55$ which is well below the percolation threshold of 0.592746 (see ...)

Diffusion Limited Aggregation Model

Witten and Sanders [3]
Growth of bacterial colonies

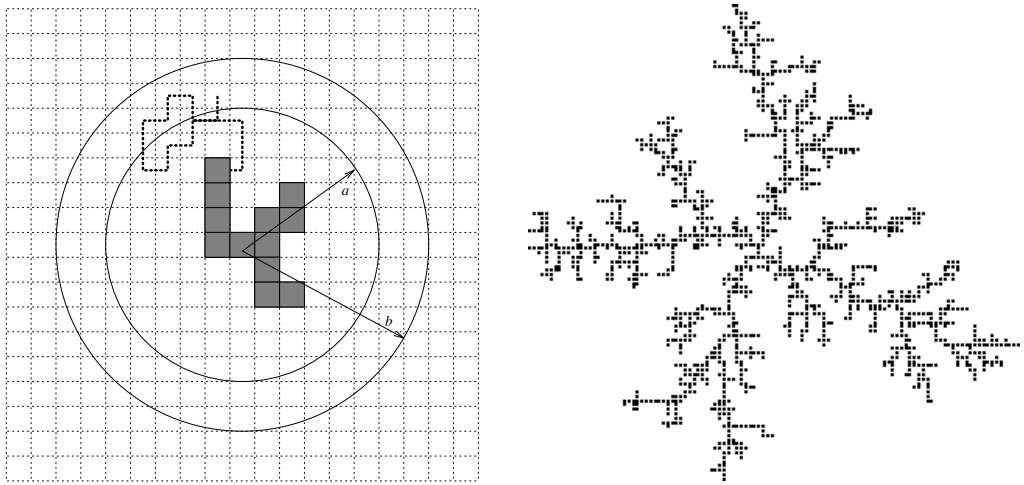

Figure: Setup procedure for the injection of a random walker for the diffusion limited aggregation (DLA). The right hand side figure shows an example of a DLA-cluster

Internal diffusion-limited aggregation (iDLA)
is a cluster growth process in which particles start at one or more sources within a cluster, diffuse outward, and are added to the cluster at the first site outside it they reach [4]

Diffusion Limited Aggregation Model

A variant of the DLA is the **Internal diffusion-limited aggregation (iDLA)**
is a cluster growth process in which particles start at one or more sources within a cluster, diffuse outward, and are added to the cluster at the first site outside it they reach [4]
ballistic growth

Algorithm 3 Basic Algorithm: DLA

choose initial site s
choose a radius R_1 around s
choose a radius R_2 around s such that $R_1 < R_2$
start a random walker at a random position on the circle with radius R_1
while *max_sites* not reached **do**
  while position of random walker within R_2 **do**
    advance the random walker one step
    if random walker is nearest neighbour of an occupied site **then**
      add site to list of occupied sites
      start a new random walker at a random position on the circle with radius R_1
    else if site is outside of R_2 **then**
      start a new random walker at a random position on the circle with radius R_1
    end if
    compute distance d of the nearest occupied site to R_1
    if $d < d_c$ **then**
      increase R_1 and R_2
    end if
  end while
end while

Diffusion Limited Aggregation Model

UNIVERSITÄT HEIDELBERG ZUKUNFT SEIT 1386

```
while (sites < max_sites):
    w  = start_rw_on_circle(r1,midx,midy)
    while (check_inside_disk(w,r2,midx,midy)):
        w  = advance_random_walk(w)
        if ( nearest_neighour_of_occupied_site(w,grid) ):
            grid[w[0]][w[1]] = 1
            sites           += 1
            cm = center_of_mass(grid,cols,rows)
            r1 = cluster_radius(cm) + d
            r2 = r1 + dx
            w  = start_rw_on_circle(r1,midx,midy)
```

Code 4: Diffusion-Limited Aggregation

Diffusion Limited Aggregation Model

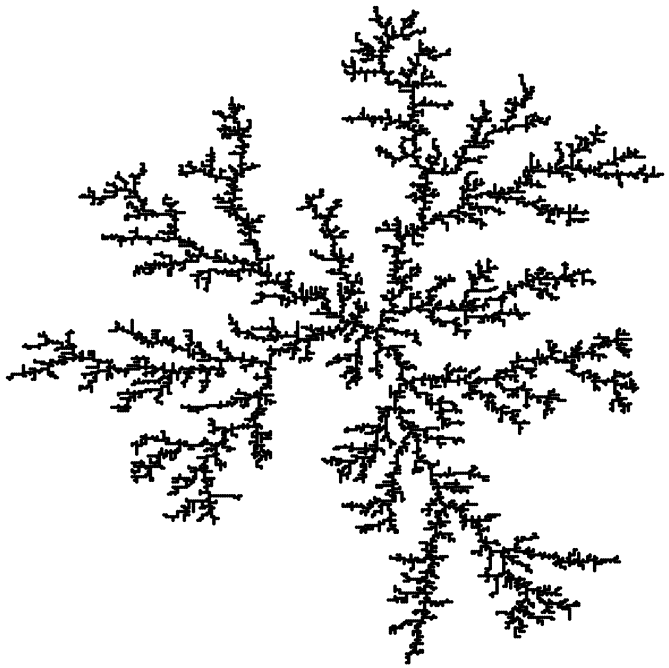

Figure: DLA-cluster with 10000 occupied sites

$$C(r, r') =< n(r)n(r + r') > \propto r^{d-d_f}$$

fractal dimension for $d = 2$ $d_f = 1.66$

Deposition

Figure:

Contact Processes I

We shall assume either a lattice $\Lambda = L^d$ or $\Lambda = \mathbf{Z}_n \times \mathbf{Z}_m$, where n and m are integers, $\Lambda = \mathbf{Z}^d$, or generally a graph $\Lambda = G$ where each node is called a site s. Let $\lambda \in (0, \infty)$. In the **contact process** [5] each site can be occupied or empty (infected, healthy):

- Occupied (Infected) sites become empty (healthy) at rate 1
- empty sites become occupied at a rate λ time the number of occupied neighbours

While on a finite set, the infection dies out eventually for any value of λ, on a d-dimensional lattice, there is a critical value λ_d [5]

$$\frac{1}{2d} \leq \lambda_c \leq \frac{2}{d} \tag{11}$$

such that

- all sites become empty for $\lambda \leq \lambda_d$
- there are sites that are occupied for $\lambda > \lambda_d$

A variation of the basic model is to assume a random environment [6]. This can for example be realized by assuming an edge dependent control parameter, i.e. we replace λ with λ_e where edge $e \in E(\Lambda)$. Thus we have a collection $(\lambda_e)_{e \in E(\Lambda)}$ of random variables.

- Shape Theorems

Exclusion Processes I

exclusion processes [7]
[8]

Cellular Automata Models of Growth I

multi-cellular biological systems (MCBS) [9]

Cluster-Cluster Aggregation I

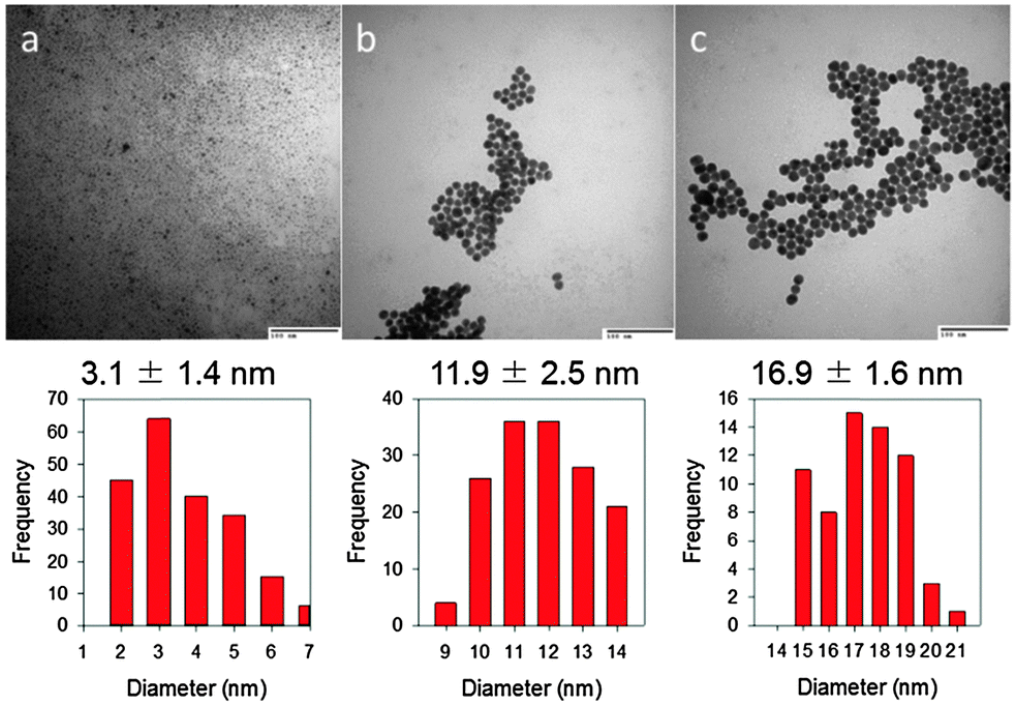

Figure: image taken from the paper *Stability, cytotoxicity and cell uptake of water-soluble dendron?conjugated gold nanoparticles with 3, 12 and 17 nm cores* by Deol et al [10] showing the aggregation of gold particles

Kolloiden

- Ausgangssituation: auf einem Gitter diffundieren Einzelteilchen

Cluster-Cluster Aggregation II

- wenn zwei Partikel sich beruehren, bilden sie Cluster aus zwei Teilchen, die ebenfalls diffundieren
- Falls ein Cluster einen anderen Cluster oder ein Einzelteilchen trifft, wird aus beiden ein groesseres Aggregat gebildet.
- Die Masse bleibt waehrend der Aggregation konstant

- $d = 2$ $d_f = 1.42$
- $d = 3$ $d_f = 1.78$

gold particle aggregation in cells [11]

Excercises I

Exercise 1: **Richardson's Model [12]**
In the Richardson's model occupied sites remain occupied. Unoccupied sites change state to occupied at time $t+1$ with probability p if at least one neighbor was occupied at time n. Show numerically that the asymptotic shape of the object (in $d=2$) has a straight edge if $p > p_c$, where p_c is a critical value. Determine p_c.

Exercise 2: **Williams and Bjerknes Tumor Growth Model [13]**
This model generalizes the Eden model as a stochastic model for the spread of cancer cells (skin cancer). At each time step a site can become either ill with probability α or healthy with probability β. Thus the ration $\kappa = \alpha/\beta$ determines the behaviour with $\kappa = \infty$ recovering the Eden model. Rewrite the above Eden model algorithm to incorporate the modification

Exercise 3: **DLA with reaction-controlled absoption**
Assume a DLA model. Let P be the probability for a particle to react with the nearest-neighbour site that is occupied. Modify the above program in incorporate the changed absorption.

Exercise 4: **Continuum Model of DLA**

In the continuum model of DLA each particle is assumed to have a radius a. At every step the random walker is performing a gaussian random walk with steps size $\leq a$. The particles sticks to the aggregate is the distance to the nearest particle is $\leq a$. Modify the above program in incorporate the changed absorption. Show that

$$d = 2 \quad d_f = 1.71 \tag{12}$$

$$d = 3 \quad d_f = 2.5 \tag{13}$$

Bibliography I

[1] Thomas M Liggett. *Stochastic Interacting Systems: Contact, Voter and Exclusion Processes*. Springer, 1999.

[2] Murray Eden. A two-dimensional growth process. volume Vol. IV, pages 223–239. Univ. California Press, Proc. 4th Berkeley Sympos. Math. Statist. and Prob, 1961.

[3] T.A. Witten und L.M. Sander. Diffusion-limited aggregation. *Physical Review B*, 27(5686), 1983.

[4] M. Bramson G. Lawler and D. Griffeath. Internal diffusion limited aggregation. *Ann. Prob.*, 20:2117–2140, 1992.

[5] T. E. Harris. Contact interactions on a lattice. pages 969–988, 1974.

[6] Richard Durrett. *Lecture notes on particle systems and percolation*. The Wadsworth and Brooks/Cole Statistics/Probability Series. Wadsworth and Brooks/Cole Advanced Books and Software, Pacific Grove, CA, 1988.

[7] Frank Spitzer. Interaction of markov processes. *Advances in Mathematics*, 5(2):246–290, 1970.

[8] Pipkin AC. MacDonald CT, Gibbs JH. Kinetics of biopolymerization on nucleic acid templates. *Biopolymers*, 6(1):1–5, 1968.

[9] Minki Hwang, Marc Garbey, Scott A Berceli, and Roger Tran-Son-Tay. Rule-based simulation of multi-cellular biological systems—a review of modeling techniques. *Cellular and molecular bioengineering*, 2(3):285–294, 09 2009.

Bibliography II

[10] Suprit Deol, Nisala Weerasuriya, and Young-Seok Shon. Stability, cytotoxicity and cell uptake of water-soluble dendron-conjugated gold nanoparticles with 3, 12 and 17 nm cores. *Journal of Materials Chemistry B*, 3(29):6071–6080, 2015.

[11] Alexandre Albanese and Warren C. W. Chan. Effect of gold nanoparticle aggregation on cell uptake and toxicity. *ACS Nano*, 5(7):5478–5489, 07 2011.

[12] D. Richardson. Random growth in a tesselation. *Proc. Camb. Phil. Soc.*, 74:515–528, 1973.

[13] Williams T and Bjerknes R. Stochastic model for abnormal clone spread through epithelial basal layer. *Nature*, 236:19–21, 1972.

Index I