

# Nukleation auf vicinalen Oberflächen

Helge Riedrich  
Fachbereich Physik



13. Juli 2006

# Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b>	<b>4</b>
<b>2</b>	<b>Grundlagen</b>	<b>5</b>
2.1	Wachstums-Arten . . . . .	5
2.2	Qualitative Betrachtung der mikroskopischen Prozesse . . . . .	5
2.3	Kritischer Keim . . . . .	6
2.4	Charakteristische Größen . . . . .	7
2.5	Zeitliche Betrachtung . . . . .	7
<b>3</b>	<b>Ratengleichungen</b>	<b>8</b>
3.1	Herleitung der Ratengleichungen . . . . .	8
3.1.1	Ein erster Ansatz . . . . .	8
3.1.2	Die Einfang-Raten . . . . .	8
3.1.3	Resultierende Gleichungen für ebene Oberflächen . . . . .	10
3.2	Bestimmung von $\sigma$ . . . . .	10
3.2.1	konstantes $\sigma$ . . . . .	10
3.2.2	geometrische Ansätze . . . . .	10
3.2.3	Uniform depletion, lattice approximation . . . . .	11
3.3	Vereinfachungen, Annahmen . . . . .	12
3.4	Zusatz-Terme für vicinale Oberflächen . . . . .	12
3.5	Resultierende Gleichungen . . . . .	14
<b>4</b>	<b>Einführende Simulationen auf ebener Oberfläche</b>	<b>15</b>
4.1	Vergleich mit ebener OF . . . . .	15
4.1.1	Phasen, Berechnung von $\sigma_x$ . . . . .	15
4.1.2	Veränderung von $\frac{D}{R}$ . . . . .	16
4.1.3	Maximale Cluster-Dichte . . . . .	17
4.1.4	mittlere Clustergröße . . . . .	17
4.2	Betrachtung einer vicinalen Oberfläche . . . . .	18
<b>5</b>	<b>Simulations-Ergebnisse</b>	<b>20</b>
5.1	Zusammenfassung anderer Ergebnisse . . . . .	20
5.2	Simulations-Ergebnisse . . . . .	21

5.3	Interpretation der Ergebnisse . . . . .	22
5.3.1	Bestimmung von $f$ . . . . .	23
5.3.2	Wert von $l_0$ , schmale Terrassen . . . . .	24
5.3.3	Ein Beispiel zur Probe . . . . .	26
5.4	weitere Simulationen . . . . .	27
5.4.1	Korrekturterme - ein Versuch . . . . .	27
5.4.2	$n_x$ bei festem $\theta$ . . . . .	29
<b>6</b>	<b>Zusammenfassung / Conclusions</b>	<b>31</b>
<b>A</b>	<b>Das Programm</b>	<b>33</b>
A.1	Die GUI . . . . .	33
A.2	Der Rechenkern . . . . .	34
A.3	Bestimmung von $\sigma$ . . . . .	34
A.4	Der ODE-Solver . . . . .	35
<b>B</b>	<b>kommentierte Programm-Texte</b>	<b>37</b>
B.1	Die GUI . . . . .	37
B.2	Der Rechenalgorithmus . . . . .	45
B.3	Der Vergleichs-Rechenalgorithmus . . . . .	47
<b>C</b>	<b>weitere Abbildungen</b>	<b>49</b>
C.1	Auf ebener Oberfläche . . . . .	49
C.2	Auf vicinaler Oberfläche . . . . .	49
	<b>Literatur</b>	<b>51</b>
	<b>Abbildungsverzeichnis</b>	<b>52</b>

# 1 Einleitung

Die Analyse von Anordnungen auf atomarer Ebenen ist ein Feld von rasch wachsender Bedeutung. So ist z.B. in der Nanotechnologie das Erstellen feinsten Strukturen notwendig, während moderne optische Elemente möglichst ebene, aber gleichzeitig extrem dünne Beschichtungen erfordern.

Zwar lassen sich einzelne Atome auch "von Hand" verschieben (z.B. mittels eines Scanning-Tunneling-Microscope), ein grundlegendes Verständnis des Verhaltens von Atomen auf einer Oberfläche könnte jedoch ganz neue Möglichkeiten eröffnen. Eine Fertigung auf der Basis von Selbstorganisation hätte gegenüber der seriellen große Vorteile, auch wenn solche Methoden weniger Kontrolle über das Endergebnis zulassen.

In dieser Arbeit soll die Struktur-Bildung von Monomeren auf einer (vicinalen) Oberfläche quantitativ betrachtet werden. Dazu wird mittels Ratengleichungen ein Aufdampf- und Diffusions-Prozess simuliert.

Während über solche Vorgänge auf ebenen Oberflächen bereits vieles bekannt ist, sind zu vicinalen Oberflächen hauptsächlich experimentelle, oder per Monte-Carlo-Simulation gewonnene Daten vorhanden.

Ein großer Nachteil solcher Modelle ist es, dass es im Vergleich zu Ratengleichungen schwerer fällt, ein kontinuierliches Spektrum an Messdaten zu ermitteln. Oft gewinnt man nur einzelne Datenpunkte, was eine Auswertung der Messungen erschwert.

Die Computer-Simulation ist inzwischen als Hilfsmittel der Physik unverzichtbar.

Dabei ist es mit Hilfe solcher Modellen nicht nur möglich, das Verhalten komplexer Systeme vorherzusehen (z.B. Wetter-, Klima-Modelle) oder die Kosten und Mühen eines aufwendigen Experimentes zu sparen.

In einigen Fällen lassen sich auch - im Abgleich mit experimentellen Ergebnissen - ganz neue Erkenntnisse über ein System gewinnen. So lassen sich z.B. Theorien über die Entstehung eines bekannten Messergebnisses oft mit Hilfe von Simulationen bestätigen oder verwerfen.

## 2 Grundlagen

In diesem Kapitel sollen die grundlegenden Prozesse eines Nukleations-Experimentes kurz geschildert werden.

Wir folgen dabei [1], [2] und [3].

### 2.1 Wachstums-Arten

Das Wachstum von Schichten lässt sich im wesentlichen in drei Arten aufteilen ([1]):

Beim Van-der-Merve-Wachstum lagern sich glatte Monomer-Schichten übereinander an. Bis auf die oberste, noch wachsende Schicht sind alle Lagen vollständig.

Im Gegensatz dazu bilden sich beim Vollmer-Weber-Wachstum dreidimensionale Cluster, die aus mehreren Lagen bestehen.

Eine Mischung aus beiden Formen findet man beim Krastanov-Wachstum. Hier entwickelt sich auf der Oberfläche zunächst eine geschlossene Monolage, auf der sich dann dreidimensionale Cluster bilden.

Die Art des Wachstums ist dabei von den Grenzflächen-Energien der beteiligten Materialien abhängig. Van-der-Merve-Wachstum wird auftreten, wenn - bei der Kondensation von Material A auf Material B - gilt:

$$\gamma_A + \gamma^* \geq \gamma_B . \quad (2.1)$$

$\gamma_A$  und  $\gamma_B$  bezeichnen dabei die freien Oberflächen-Energien der Materialien,  $\gamma^*$  die Grenzflächenenergie zwischen ihnen.

### 2.2 Qualitative Betrachtung der mikroskopischen Prozesse

Wir betrachten die Prozesse auf einer vicinalen Oberfläche, auf der Atome auftreten (Ziffern in Klammern beziehen sich auf Abb. 2.1):

Die mit Fluss-Rate  $R$  auftreffenden Monomere (1) diffundieren auf der Oberfläche (2). Treffen sie dabei auf ein anderes Monomer, können sie zu einem Cluster nukleieren (3). Treffen sie auf einen bestehenden Cluster (4), oder auf eine Stufen-Barriere der Terrassen-förmigen Oberfläche (5), lagern sie sich dort an. Außerdem können die Monomere von der Oberfläche evaporieren (6).

Ein Cluster kann - ebenso wie er Atome einfangen und dadurch wachsen kann - Atome verlieren und zerfallen (7). Beim Wachstum ist es möglich, dass ein Cluster mit einem anderen zusammenwächst (8), bzw. auf eine Stufe trifft.

Betrachtet man ein ankommendes Monomer, ist auch denkbar, dass es direkt auf einem anderen Monomer, bzw. einem Cluster landet. Auch ein direktes

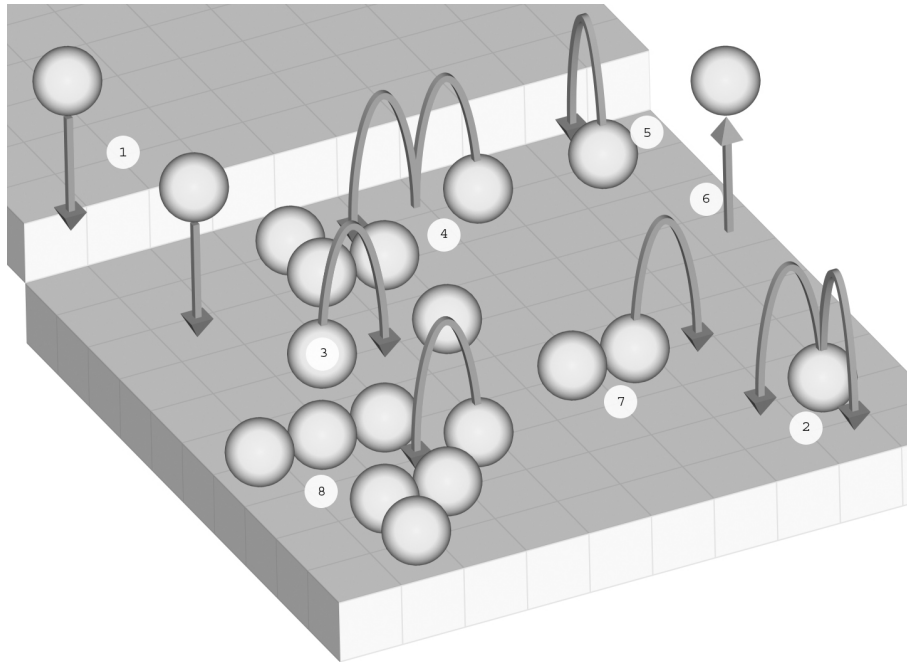


Abbildung 2.1: Schematische Darstellung von atomaren Prozessen auf vicinaler Oberfläche.

Auftreffen an einer Stufe ist möglich.  
 Auf andere Prozesse, wie z.B. die Diffusion in das Substratmaterial hinein, soll hier nicht näher eingegangen werden.

### 2.3 Kritischer Keim

Wie eben beschrieben kann ein Cluster ebenso wachsen, wie auch Atome verlieren.

Mit wachsender Cluster-Größe wird ein Zerfall unwahrscheinlicher. Die Auftreff-Wahrscheinlichkeit diffundierender Monomere wächst mit der Größe des Clusters. Dieses sorgt zusätzlich dafür, dass große Cluster weiter wachsen. Der kritische Keim  $i$  gibt dabei die Anzahl der Atome in dem größten Cluster an, bei dem noch ein Zerfall zu erwarten ist. So sind z.B. für  $i = 1$  Dimere stabil, für  $i = 2$  Trimere, usw. Im Fall  $i = 0$  friert das gesamte System - also auch Monomere - ein.

## 2.4 Charakteristische Größen

Die Diffusion der Monomere ist durch einen temperaturabhängigen Diffusions-Koeffizienten  $D$  bestimmt:

$$D \propto \alpha \cdot a^2 \cdot \nu_0 \cdot e^{-E_D \beta} . \quad (2.2)$$

$\alpha \approx \frac{1}{4}$  ist eine Konstante,  $\nu_0$  bezeichnet die Versuchs-Frequenz und liegt im Bereich  $10^{13}$ . Die Diffusions-Energie  $E_D$  liegt in der Größenordnung einiger zehntel eV,  $\beta$  bezeichnet den Faktor  $\frac{1}{k_B T}$ .

Bei konstanter Depositionsrate  $R$  gilt für die Bedeckung  $\theta$ :

$$\theta = R \cdot t . \quad (2.3)$$

Im sogenannten Submonolagen-Regime, also für  $\theta < 1$  betrachten wir vor allem die Monomer-Dichte  $n_1$  und die Dichte  $n_j$  von Clustern der Größe  $j$ . Für die gesamt-Cluster-Dichte  $N$  gilt also:

$$N = \sum_{j \geq 2} n_j(\theta) . \quad (2.4)$$

Die durchschnittliche Clustergröße lässt sich bestimmen aus der gesamt-Zahl der abgelagerten Monomere, der Monomer-Dichte und Cluster-Dichte:

$$\langle j \rangle = \frac{\theta - n_1 - n_s}{N} . \quad (2.5)$$

$n_s$  bezeichnet die Bedeckung durch Monomere, die an Stufen angelagert sind, so dass  $\theta - n_1 - n_s$  die Bedeckung durch Cluster darstellt.

## 2.5 Zeitliche Betrachtung

Im Submonolagen-Bereich lassen sich drei Phasen unterscheiden ([2]).

Anfänglich ist die Monomer-Dichte wesentlich höher, als die Cluster-Dichte. In dieser Nukleations-Phase überwiegt die Bildung von Dimeren. Die Zahl der Cluster wächst rasch an, während ihre durchschnittliche Größe nur geringfügig steigt.

Mit wachsender Cluster-Dichte treffen die Monomere immer häufiger auf Cluster. Diese Sättigungs-Phase zeichnet sich durch eine annähernd konstante Cluster-Dichte aus, die durchschnittliche Cluster-Größe wächst merklich. In einer dritten Phase ist das Zusammenwachsen von Clustern der dominierende Prozess. Die Cluster-Dichte sinkt bei rasch steigendem  $\langle j \rangle$ .

### 3 Ratengleichungen

Ziel dieser Arbeit ist eine Darstellung von Nukleations-Abläufen auf vicinalen Oberflächen. Dabei soll vor allem die Clusterdichte in Abhängigkeit von Bedeckung, Diffusionsrate und Terrassenbreite untersucht werden.

Im Gegensatz zu anderen Arbeiten (z.B. [4], [5]) soll dieses jedoch nicht mit einer Monte-Carlo-Simulation, sondern auf der Basis von Ratengleichungen geschehen.

Bei der Herleitung folgen wir [6] und [1] - diese befassen sich jedoch mit ebenen Oberflächen, so dass später entsprechende Korrektur-Glieder eingefügt werden müssen.

#### 3.1 Herleitung der Ratengleichungen

##### 3.1.1 Ein erster Ansatz

Wir betrachten zuerst die Dichten  $n_1$  der Monomere,  $n_j$  der Cluster der Größe  $j$  und  $n_x$  der stabilen Cluster ( $j > i$ ):

$$\frac{dn_1}{dt} = R - \frac{n_1}{\tau_a} - 2U_1 - U_x - \sum_{j>1}^i U_j . \quad (3.1a)$$

$$\frac{dn_j}{dt} = U_{j-1} - U_j, (1 < j \leq i) . \quad (3.1b)$$

$$\frac{dn_x}{dt} = U_i - \dots . \quad (3.1c)$$

Monomere werden mit Depositionrate  $R$  auf die Oberfläche aufgebracht und evaporieren mit Rate  $\frac{n_1}{\tau_a}$ .  $U_n$  bezeichnet die Einfang-Rate eines Clusters der Größe  $n$ . Bei der Bildung von Dimeren (mit Einfang-Rate  $U_1$ ) gehen dem System je zwei Monomere verloren, bei der Anlagerung an Clustern der Größe  $j$  je eines.  $U_x$  beschreibt die Einfang-Rate stabiler Cluster, also mit  $j > i$  ( $i$  entspricht dem kritischen Keim, siehe Abschnitt 2.3).

Die Dichte stabiler Cluster  $n_x$  wächst, wenn sich an einen Cluster der Größe  $i$  ein Monomer anlagert. An dieser Stelle müssen Korrektur-Terme eingefügt werden, die z.B. das Zusammen-Wachsen von Clustern beschreiben.

##### 3.1.2 Die Einfang-Raten

Für die Einfang-Rate eines stabilen Clusters gilt:

$$U_x = \sum_k (\sigma_k D n_1 n_x + R a_k n_k) . \quad (3.2)$$

Dabei geben  $\sigma_k$  und  $a_k$  den Einfang-Querschnitt bzw. die Fläche eines Clusters der Größe  $k$  an.  $\sigma_k D n_1 n_x$  beschreibt die Anlagerung per Diffusion,  $R a_k n_k$  die Anlagerung von Monomeren, die bei der Deposition direkt an, bzw. auf einem Cluster landen.  $D$  ist der Diffusionskoeffizient von Monomeren.

Mit entsprechenden Durchschnittswerten  $\sigma_x$  und  $a_x$  lässt sich Glg. 3.2 vereinfachen zu:

$$U_x = \sigma_x D n_1 n_x + R a_x n_x . \quad (3.3)$$

Analog ergibt sich für  $U_1$ :

$$U_1 = \sigma_1 D n_1^2 + R a_1 n_1 . \quad (3.4)$$

Für die Entstehung stabiler Cluster gilt:

$$U_i = \sigma_i D n_1 n_i + \sum_k \sum_l \sigma_{kl} n_k n_l (D_k + D_l) . \quad (3.5)$$

Der erste Term entspricht der Anlagerung eines Monomers an einen Cluster der kritischen Größe  $i$ , der zweite Term beschreibt das Aufeinandertreffen zweier Cluster der Größe  $l$  und  $k$  mit  $l + k > i$ . Die Cluster-Diffusionskoeffizienten  $D_k$  und  $D_l$  sind deutlich kleiner als  $D$ .

Ein weiterer zu beachtender Prozess ist das Zusammen-Wachsen von Clustern. Ein für kleine Cluster zufriedenstellendes Ergebnis liefert der Ansatz ([2]):

$$U_c = 2 n_x \frac{dZ}{dt} . \quad (3.6)$$

$Z$  beschreibt den von von stabilen Clustern bedeckten Teil der Fläche.

Da aber nicht nur Monomere, sondern auch Cluster diffundieren, führt dieses zu einer weiteren Veränderung der Cluster-Dichte:

$$U_m = \frac{1}{2} \sum_k \sum_l \sigma_{kl} n_k n_l (D_k + D_l) . \quad (3.7)$$

Die Beschreibung ist dabei analog zum zweiten Term von  $U_i$ , allerdings in diesem Fall mit  $k, l > i$ .

Für eine weitere Vereinfachung müssen Annahmen über die Diffusionskoeffizienten  $D_k$  und  $D_l$  gemacht werden. Es bieten sich z.B. Modelle an, bei denen  $D_j$  unabhängig von der Größe  $j$  der Cluster ist, oder bei denen sich nur Cluster bis zu einer bestimmten Größe  $j_{max}$  bewegen.

### 3.1.3 Resultierende Gleichungen für ebene Oberflächen

$$\frac{dn_1}{dt} = R - \frac{n_1}{\tau_a} - 2(\sigma_1 D n_1^2 + R a_1 n_1) - (\sigma_x D n_1 n_x + R a_x n_x). \quad (3.8)$$

$$\frac{dn_x}{dt} = \sigma_i D n_1 n_i + \sum_k \sum_l \sigma_{kl} n_k n_l (D_k + D_l) - 2n_x \frac{dZ}{dt} - \frac{1}{2} \sum_k \sum_l \sigma_{kl} n_k n_l (D_k + D_l). \quad (3.9)$$

Dabei ist bei den beiden Doppel-Summen zu unterscheiden, dass die erste über  $k, l < i; k + l > i$  und die zweite über  $k, i > l$  laufen.

## 3.2 Bestimmung von $\sigma$

Zur Bestimmung der Einfang-Querschnitte  $\sigma_1$  und  $\sigma_x$  existieren verschiedene Ansätze, von denen vier der bekannteren hier kurz angesprochen werden sollen. Bei allen der genannten Verfahren handelt es sich um Näherungen, die bestenfalls ungefähr den realen Gegebenheiten entsprechen. Komplexere Versuche, die effektiven Querschnitte zu errechnen, können zwar im Einzelfall die Genauigkeit erhöhen, aber keine absolut korrekten Ergebnisse liefern ([2]).

### 3.2.1 konstantes $\sigma$

In der einfachsten Variante werden beide Werte als konstant angenommen. Im Punktinsel-Modell ([3]) vernachlässigt man die räumliche Ausdehnung von Clustern und setzt  $\sigma_1 = \sigma_x = 1$ .

Venables schlägt für  $\sigma_i$  ( $i \geq 1$ ) Werte zwischen 2 und 4 vor,  $\sigma_x$  soll im Bereich von 5 bis 10 liegen ([1]). In den Anfängen der Nukleations-Phase liefert diese Näherung auch - bei geschickter Wahl der Konstanten - recht gute Werte. Spätestens zum Beginn der Sättigungs-Phase, also mit steigender Clustergröße, ist diese Näherung jedoch nicht mehr sinnvoll.

### 3.2.2 geometrische Ansätze

Ein anderer Ansatz stützt sich auf Clustergröße und -Geometrie. Venables geht von einer Proportionalität zwischen Anzahl der Monomere  $j$  in einem Cluster und deren Querschnitt aus. Bei dreidimensionalem Wachstum mit  $\sigma_j \propto j^{1/3}$ , bei zweidimensionalen analog  $\sigma_j \propto j^{1/2}$ .

Einen verfeinerten Ansatz liefern Überlegungen zur Geometrie von zweidimensionalen, fraktalen Clustern ([7]).

$$\sigma_j \propto j^{\frac{1}{H}}. \quad (3.10)$$

Dabei steht  $H$  für die fraktal-Dimension der Cluster. Einen anderen Ansatz stellt die Addition einer Konstanten zu einem solchen Term ausdar:

$$\sigma_j = 2 + (j)^{\frac{1}{H}} . \quad (3.11)$$

Der Parameter  $H$  ist von der geometrischen Anordnung der Cluster abhängig, für eine fraktale Clusterform sollte dieser Werte z.B. im Bereich von 1, 7 liegen ([2]). Setzt man die durchschnittliche Clustergröße (Glg. 2.5) ein, ergibt sich:

$$\sigma_x = 2 + \langle x \rangle^{\frac{1}{H}} . \quad (3.12)$$

### 3.2.3 Uniform depletion, lattice approximation

Für eine noch bessere Beschreibung dieser Querschnitte lassen sich Bessel-Funktionen einsetzen ([6],[1]).

$$\sigma_j = 2\pi X_j \frac{K_1(X_j)}{K_0(X_j)} . \quad (3.13)$$

Dabei ist  $X_j^2 = \frac{r_j^2}{D\tau}$ ,  $r_j$  beschreibt den Radius des Clusters. Die Verweildauer  $\tau$  setzt sich aus einem Evaporationsterm und einem Anlagerungsterm zusammen:  $\tau^{-1} = \tau_a^{-1} + \tau_c^{-1}$ .  $\tau_c = D\sigma_x n_x$  entspricht der Verweildauer vor der Anlagerung an einen stabile Cluster. Setzt man die durchschnittliche Clustergröße ein, ergibt sich mit  $X^2 = \frac{r_x^2}{D\tau}$ :

$$\sigma_x = 2\pi X \frac{K_1(X)}{K_0(X)} . \quad (3.14)$$

Im Falle einer kompletten Kondensation, also  $\tau_a^{-1} = 0$ , lässt sich  $X$  als Funktion von  $Z = \pi n_x r_x^2$  auffassen:

$$X^2 = \frac{\sigma_x Z}{\pi} . \quad (3.15)$$

Die lattice approximation berechnet  $\sigma_x$  ebenfalls als Funktion von  $Z$ . Für  $\tau_a^{-1} = 0$  lässt sie sich wie folgt darstellen:

$$\sigma_x = 8\pi \frac{1 - Z}{-2 \ln(Z) - (3 - Z)(1 - Z)} . \quad (3.16)$$

Einen Vergleich dieser Berechnungsmethoden zeigt Abb. 3.1. Dabei ist die mittels uniform depletion erzeugte Kurve als untere, die mittels lattice approximation erzeugte als obere Grenze der zu erwartenden Werte anzusehen. Für die folgenden Simulationen wurden die  $\sigma_x$ -Werte mit der Uniform-Depletion-Näherung bestimmt.

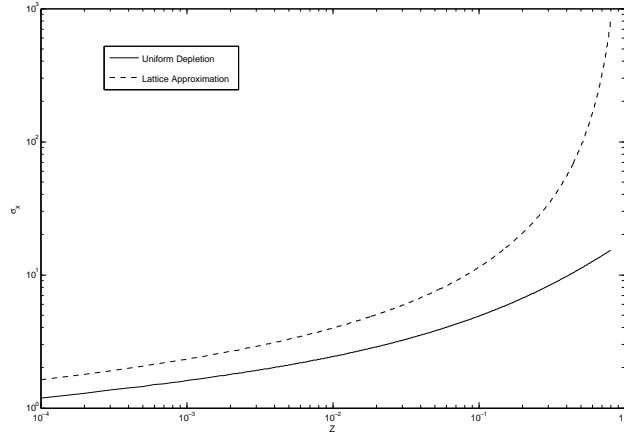


Abbildung 3.1: Vergleich zwischen Uniform Depletion- und Lattice-Approximation, analog zu [6]

### 3.3 Vereinfachungen, Annahmen

Als Grundlage der Simulation gehen wir von folgendem Spezialfällen aus:

- Wir betrachten Van-der-Merve-Wachstum, also eine Schicht-für-Schicht Anlagerung.
- Monomere-Evaporation findet nicht statt, Teilchen, die auf der Oberfläche auftreffen, bleiben, der Term  $\frac{n_1}{\tau_a}$  entfällt. Außerdem lässt sich in diesem Fall die Querschnittsberechnung der Bessel-Funktion vereinfachen: Mit  $\tau^{-1} = \tau_c^{-1} = Dn_x\sigma_x$  ergibt sich  $X = r_x^2 n_x \sigma_x = \frac{Z}{\pi} \sigma_x$ .
- Der kritische Keim sei  $i = 1$ , Cluster aus 2 Molekülen sind also bereits stabil und zerfallen nicht wieder. Die Variablen  $\sigma_i$  und  $n_i$  in Glg. 3.9 sind durch  $\sigma_1$  und  $n_1$  zu ersetzen.
- Der Diffusions-Koeffizient für Cluster sei 0, Cluster bewegen sich nicht. Die beiden Doppelsummen in Glg. 3.9 verschwinden also.

Dieser Spezialfall ist zum einen vergleichsweise gut zu beschreiben, zum anderen sind zu diesem Fall einige Vergleichs-Werte vorhanden.

### 3.4 Zusatz-Terme für vicinale Oberflächen

Für die Betrachtung einer vicinalen Oberfläche sind weitere Terme notwendig, die die Anlagerung an die Stufen-Grenzen beschreiben. Zuerst definie-

ren wir die dimensionslose Stufenbreite  $l$  als Quotient aus eigentlicher Stufenbreite  $\omega$  und dem Monomer-Durchmesser  $a$ . Indem wir die Stufenbreite durch die Anzahl der nebeneinander passenden Monomere beschreiben, können wir die Vorgänge auf den Terrassen unabhängig von der Größe der aufgebrachten Atome beschreiben. Für die Anlagerung an diese Stufen ergibt sich ein Zusatz-Term  $-\frac{D}{l}n_1$ . Gehen wir davon aus, dass Monomere, die beim Auftreffen im Querschnitt über bzw. an einer Kante liegen, ergibt sich - analog zur direkten Anlagerung auf Clustern - der Term  $-\frac{R}{l}$ . Nimmt man die bereits dort angelagerten Stufen-Atome  $n_s$  hinzu, erweitert sich dieser Term zu  $-R(\frac{1}{l} + n_s)$ . Weitere Effekte, wie z.B. das Anwachsen von Clustern an Stufen, oder die Verkleinerung des Einfang-Querschnitts von Clustern durch die direkte Nähe einer Stufe sind schwerer darzustellen. Während sie bei Monte-Carlo-Simulationen durch ein starres Gitter noch zu beschreiben sind, wären in Ratengleichungen auch Abstände zu berücksichtigen, die nicht aus ganzen Atomradien bestehen. Solche Effekte, wie auch der durch Anlagerung veränderten Form der Stufen, werden allerdings erst bei fortgeschrittener Simulation deutliche Auswirkungen zeigen. Wir betrachten daher vor allem den Nukleations-Beginn bis ca.  $\theta = 0,3ML$  ( $ML$  entspricht Monolagen).

Eine gravierende Änderung ergibt sich jedoch für das Zusammenwachsen von Clustern. Die Abdeckung durch Cluster, die sich bei ebenen Oberflächen (und  $i = 1$ ) noch durch  $Z = \theta - n_1$  beschrieben ließ, erweitert sich um einen Term für die Stufen-Atome:

$$Z = \theta - n_1 - n_s , \quad (3.17a)$$

und daraus:

$$\frac{dZ}{dt} = \frac{d\theta}{dt} - \frac{dn_1}{dt} - \frac{dn_s}{dt} = R - \frac{dn_1}{dt} - \frac{dn_s}{dt} . \quad (3.17b)$$

Ohne diese Umformung würde das Anwachsen durch neue Stufen-Atome zwar die Cluster-Zahl, nicht aber die Bedeckung durch Cluster reduzieren, die durchschnittliche Clustergröße würde überschätzt. Durch diese Umformung vernachlässigen wir das Heranwachsen von Clustern an Stufen. Da aber auch zu einer Seite angewachsene Clustern im Verhalten noch eher Clustern, als Stufen-Atomen ähneln, scheint diese Beschreibung sinnvoller. Außerdem wird die Bedeutung dieses Prozessen erst in Bereichen größerer Abdeckung zu merklichen Änderungen führen.

Ein weiterer Effekt, der hier nicht berücksichtigt wird, ist die Verschmälerung von Terrassen durch an Stufen angelagerte Monomere.

### 3.5 Resultierende Gleichungen

Es ergeben sich die folgenden Differentialgleichungen:

$$\frac{dn_1}{dt} = R - 2(\sigma_1 D n_1^2 + R a_1 n_1) - (\sigma_x D n_1 n_x + R a_x n_x) - R \frac{D}{l} - R a_1 n_s - D \frac{n_1}{l} . \quad (3.18a)$$

$$\frac{dn_x}{dt} = \sigma_1 D n_1^2 + R a_1 n_1 - 2n_x \left( R - \frac{dn_1}{dt} - \frac{dn_s}{dt} \right) . \quad (3.18b)$$

Um die spätere Auswertung zu vereinfachen, stellen wir die Gleichungen noch nach  $\theta = Rt$  um. Da  $n_1$  und  $n_s$  bereits in  $ML$  berechnet werden, ersetzen wir noch  $a_x n_x$  durch  $Z$ . Außerdem führen wir das oben genannte  $\frac{dn_s}{d\theta}$  als dritte Gleichung ein:

$$\frac{dn_1}{d\theta} = \frac{l-1}{l} - 2n_1 - Z - n_s - \frac{D}{R} \left( 2\sigma_1 n_1^2 + \sigma_x n_1 n_x + \frac{n_1}{l} \right) . \quad (3.19a)$$

$$\frac{dn_x}{d\theta} = \frac{D}{R} \sigma_1 n_1^2 + n_1 - 2n_x \left( 1 - \frac{dn_1}{d\theta} - \frac{dn_s}{d\theta} \right) . \quad (3.19b)$$

$$\frac{dn_s}{d\theta} = \frac{1}{l} + n_s + \frac{D}{R} \frac{n_1}{l} . \quad (3.19c)$$

Vernachlässigt man die Terme für direkte Anlagerung, stellen sich die Gleichungen wie folgt dar:

$$\frac{dn_1}{d\theta} = 1 - \frac{D}{R} \left( 2\sigma_1 n_1^2 + \sigma_x n_1 n_x + \frac{n_1}{l} \right) . \quad (3.20a)$$

$$\frac{dn_x}{d\theta} = \frac{D}{R} \sigma_1 n_1^2 - 2n_x \left( 1 - \frac{dn_1}{d\theta} - \frac{dn_s}{d\theta} \right) . \quad (3.20b)$$

$$\frac{dn_s}{d\theta} = \frac{D}{R} \frac{n_1}{l} . \quad (3.20c)$$

Dabei ergibt sich  $D$  aus Glg. 2.2,  $\sigma_x$  aus Glg. 3.14.

Im weiteren werden wir vor allem Gleichungen 3.20a, 3.20b und ?? verwenden. Zwar zeigen sich ähnliche Verläufe von  $n_1$  und  $n_x$  wie bei 3.19a, 3.19b und ??, bei empfindlichen Interpretationen sind die Ergebnisse jedoch leichter verwendbar, da man auf zusätzliche Korrekturen (z.B. der Depositionsrate) verzichten kann.

## 4 Einführende Simulationen auf ebener Oberfläche

Dieses Kapitel soll zunächst einige einführende Simulationen darstellen, um Abläufe transparenter zu machen und die Ergebnisse mit bekannten Gesetzmäßigkeiten abzugleichen.

### 4.1 Vergleich mit ebener OF

Zunächst betrachten wir Daten, die durch die Simulation ebener Oberflächen gewonnen wurden. Indem man von unendlicher Terrassenbreite ausgeht, also alle Terme der Form  $\frac{1}{l}$  wegfällt, lassen sich mathematische Umformungen der Ratengleichungen vermeiden. Außerdem erscheint es anschaulich, dass eine Terrasse unendlicher Ausdehnung einer Ebene entspricht.

Die Betrachtung von stufenfreien Oberflächen bietet zwei Vorteile:

Zum einen lässt sich das System hauptsächlich auf die Variable  $\frac{D}{R}$  beschränken, zum anderen ist mehr über solche Systeme bekannt, was Vergleiche erleichtert. Wir wollen hier im wesentlichen auf drei Dinge eingehen: Zuerst betrachten wir den Verlauf von  $n_1$  und  $n_x$  in einer Nukleations-Simulation, um den grundlegenden Ablauf darzustellen. In einem zweiten Abschnitt beschäftigen wir uns mit variablem  $\frac{D}{R}$ . Auch hier sollen im wesentlichen die Veränderungen qualitativ dargestellt werden. Zum Schluss werden noch einige Vergleiche mit bekannten Ergebnissen gezogen.

#### 4.1.1 Phasen, Berechnung von $\sigma_x$

Zum Vergleich betrachten wir zunächst die Entwicklung von  $n_x$  mit durch verschiedene Verfahren berechnete  $\sigma$ s.

Während anfänglich die Kurven annähernd übereinstimmen, zeigen sich im späteren Verlauf deutliche Unterschiede (Abb. 4.1). Im weiteren Verlauf verwenden wir die Berechnung mittels Bessel-Funktionen, die tatsächlich beobachtete Entwicklung von den genannten Verfahren am besten beschreibt ([6]).

Gut zu erkennen ist bei der mit Bessel-Funktionen berechneten Kurve ist auch die Phasen-Einteilung (Abschnitt 2.5). Während im linken Abschnitt (von Abb. 4.2) bei großer Monomer-Dichte die Bildung von Dimeren der beherrschende Vorgang ist, setzt sich die Anlagerung an Cluster im zweiten Sektor durch. Im Sättigungsbereich gleichen sich Neubildung und Zusammenwachsen von Clustern aus. Im letzten Abschnitt - hier kaum abgebildet - sinkt die Cluster-Dichte wieder. In diesem Bereich wachsen mehr Inseln zusammen als neu entstehen.

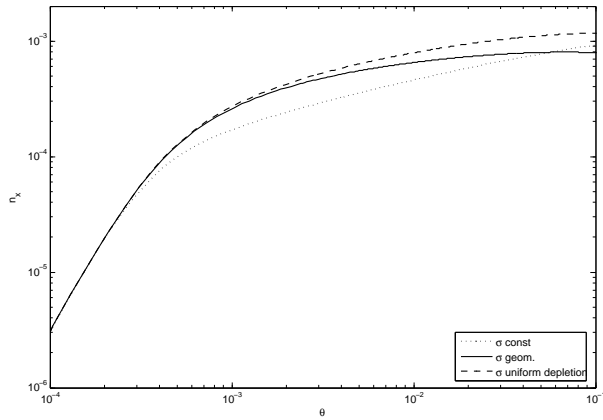


Abbildung 4.1: Vergleich der Berechnungsmethoden von  $\sigma_x$  für eine Abdeckung von  $10^{-4}$  bis  $10^{-1} ML$

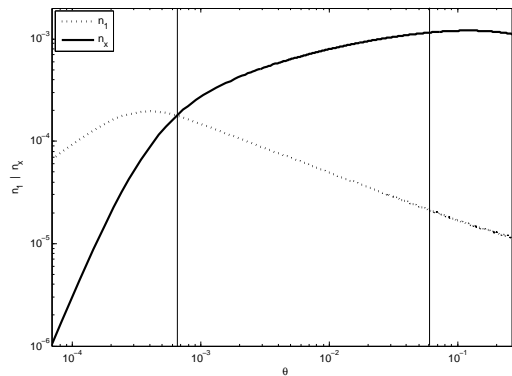


Abbildung 4.2: Einteilung in Nukleations-Phasen

#### 4.1.2 Veränderung von $\frac{D}{R}$

Um die Auswirkungen einer Veränderung von  $D$  zu verdeutlichen betrachten wir auf ebener Oberfläche den Werte-Bereich  $10^2 - 10^6$ .  $R$  ist hierbei konstant 0, 1.

Abb. 4.3 zeigt die Clusterdichte für verschiedene Diffusions-Koeffizienten. Dabei ist zu erkennen, dass die Clusterdichte für kleine  $D$  langsamer steigt, die maximal-Dichte wird erst bei größerer Bedeckung erreicht. Dieses Maximum ist dabei größer, als für kleinere  $D$ . Die - im Vergleich zur Diffusionsrate - größere Depositionsrate sorgt hier für mehr verfügbare Monomere, die

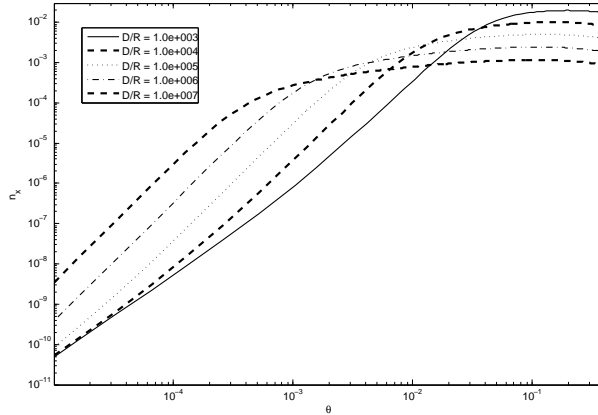


Abbildung 4.3: Clusterdichte für verschiedene  $\frac{D}{R}$  in Abhängigkeit von  $\theta$

neue Cluster bilden können. Im anderen Fall ist durch die starke Diffusion die Anlagerung an bereits vorhandene Cluster der dominierende Prozess, die Dichte der verfügbaren Monomere bleibt gering.

#### 4.1.3 Maximale Cluster-Dichte

Auf ebenen Oberflächen gilt für die maximale Cluster-Dichte und ausreichend hohe  $\frac{D}{R}$  ([5]):

$$n_x^{max} \propto \left(\frac{D}{R}\right)^{-\frac{i}{i+2}} = \left(\frac{D}{R}\right)^{-\frac{1}{3}}. \quad (4.1)$$

Tragen wir die beiden Werte einer Simulation auf ebener Oberfläche gegeneinander auf, ergibt sich die erwartete Gerade (Abb. 4.4).

#### 4.1.4 mittlere Clustergröße

Wir betrachten das Modell im Anfangsbereich der Nukleation. Abb. 4.5 zeigt die durchschnittliche Clustergröße für in Abhängigkeit von der Bedeckung. Zu Beginn ist die Cluster-Größe fast konstant. Hier ist die Bildung von Dimeren der beherrschende Prozess, die Anlagerung an bestehende Cluster spielt nur eine untergeordnete Rolle. Im weiteren Verlauf steigt die Zahl der Atome pro Cluster deutlich an. Dabei ist vor allem die in der Aggregationsphase (vgl. Abb. 4.2, rechter Block) geltende Beziehung  $\langle x \rangle \propto \left(\frac{D}{R}\right)^{\frac{1}{3}}\theta$  ([3]) gut erkennbar.

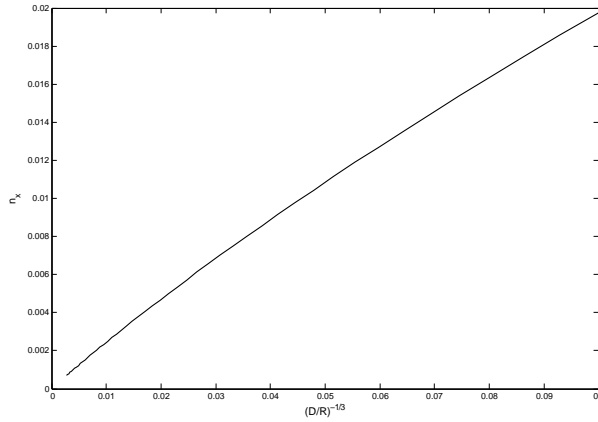


Abbildung 4.4: Maximale Cluster-Dichte in Abhängigkeit von  $(\frac{D}{R})^{-\frac{1}{3}}$

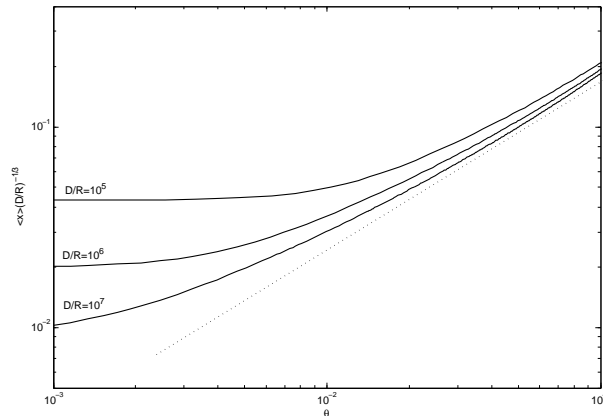


Abbildung 4.5: Durchschnittliche Clustergröße in Abhängigkeit von  $\theta$  für  $\frac{D}{R} = 10^5; 10^6; 10^7$

## 4.2 Betrachtung einer vicinalen Oberfläche

An dieser Stelle betrachten wir erstmals die Simulation einer vicinalen Oberfläche. Analog zu 4.1 wollen wir auch hier zunächst qualitative Betrachtungen durchführen.

Abb. 4.6 stellt den Verlauf der Cluster-Dichte für Stufenbreiten von 64 und 128 im Vergleich zur ebenen Oberfläche dar. Es ist gut zu erkennen, dass mit wachsender Terrassenbreite die maximale Cluster-Dichte zunimmt und diese früher erreicht wird. Analog zu [4] betrachten wir die maximale Cluster-Dichte in Abhängigkeit von der Terrassen-Breite in Abb. 4.7. Dabei werden

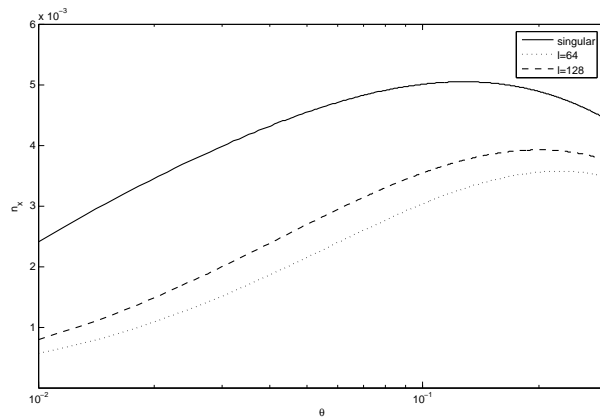


Abbildung 4.6: Darstellung der Clusterdichte bei  $\frac{D}{R} = 10^5$

jeweils nur die Terrassen-Breiten berücksichtigt, bei denen die maximale Clusterdichte spätestens bei  $\theta = \frac{1}{3}ML$  erreicht ist.

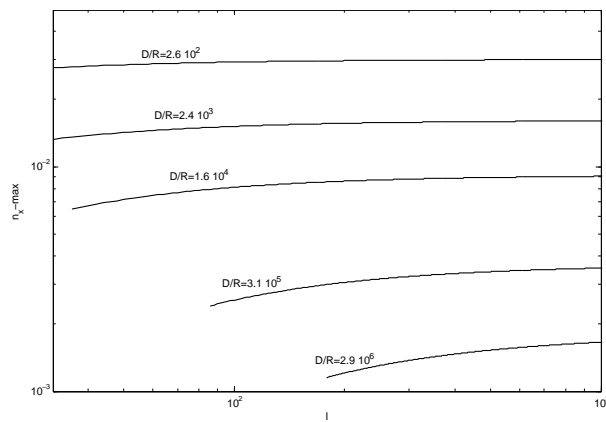


Abbildung 4.7: Maximale Cluster-Dichte in Abhängigkeit von der Terrassenbreite für verschiedene  $\frac{D}{R}$

## 5 Simulations-Ergebnisse

Das Hauptaugenmerk liegt hier auf der Beziehung zwischen der Terrassenbreite und der maximalen Clusterdichte.

Dafür betrachten wir zunächst Ergebnisse anderer Autoren ([4], [8], [5]).

Im nächsten Schritt präsentieren wir Simulationsergebnisse, die dann interpretiert werden.

### 5.1 Zusammenfassung anderer Ergebnisse

Nachdem bislang hauptsächlich Grundlagen zu ebenen Oberflächen erörtert wurden, wollen wir uns hier dem Cluster-Wachstum auf vicinalen Oberflächen widmen. Während auf ebenen Oberflächen noch recht einfache Gesetzmäßigkeiten erkennbar waren (z.B. Glg. 4.1), lassen sich auf Terrassen zwei grundsätzliche Fälle unterscheiden:

Ist die Diffusions-Länge  $l_d$  kleiner als die Terrassen-Breite, ändert sich die Clusterdichte nur unwesentlich. Dabei führen neben breiten Stufen auch große Abdeckung (durch geringe Abstände zwischen den Clustern) und ein niedriges  $\frac{D}{R}$  dazu, dass sich das Verhalten dem auf ebenen Oberflächen annähert ([5]). Im Falle größerer  $l_d$  gilt laut Pimpinelli:

$$n_x \propto l^i \left(\frac{D}{R}\right)^{-\frac{i}{2}}. \quad (5.1)$$

Für  $i = 1$  erhielt Bales für die frühen Nukleations-Stadien ([5]):

$$n_x \propto \left(\frac{D}{R}\right)^{-1} l^3 \frac{K_1\left(\frac{\sqrt{12}}{l}\right)}{K_0\left(\frac{\sqrt{12}}{l}\right)} \theta. \quad (5.2)$$

Eine Verallgemeinerung für  $i \geq 1$  ergibt sich mit:

$$n_x \propto l^{2(i+1)} \left(\frac{D}{R}\right)^{-i} \theta. \quad (5.3)$$

In einem ersten Schritt vergleichen wir in die bekannten Ergebnisse mit der maximalen Cluster-Dichte aus der Simulation (Abb. 5.1 für Glg. 4.1 und Glg. 5.2). Zwar stimmen beide Ausgleichskurven ( $\propto D^{-\frac{1}{3}}$  und  $\propto D^{-1}$ ) für kleine Bereiche von  $\frac{D}{R}$  mit der Simulationskurve überein, jedoch weichen sie zu beiden Seiten stark von den Zielwerten ab. Es lässt sich klar erkennen, dass eine Näherung der Form  $n_x \propto \left(\frac{D}{R}\right)^y$  je nur für kleine Bereiche von  $\frac{D}{R}$  zufriedenstellende Ergebnisse liefern kann.

Ähnlich verhält es sich mit einer Proportionalität zu  $l^y$ .

Zum Vergleich legen wir Ausgleichskurven der proportional zu  $l^1$  (Glg 5.1) und  $l^2$  an einen Datenverlauf ( $\frac{D}{R} = 2.9 \cdot 10^6$ ) aus Abb. 4.7 an (Abb. 5.2).

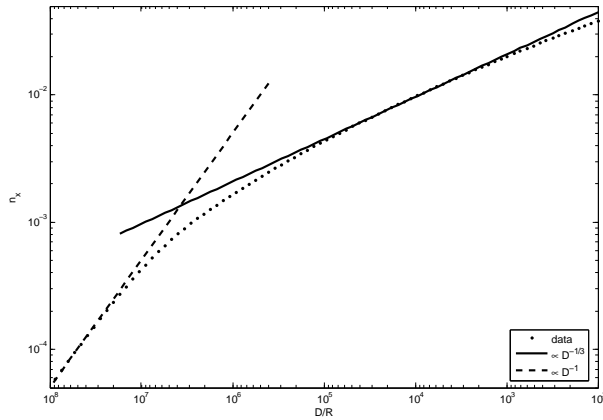


Abbildung 5.1: Vergleich zwischen Simulations-Ergebnissen und vorgegebenen Formeln für  $l = 128$

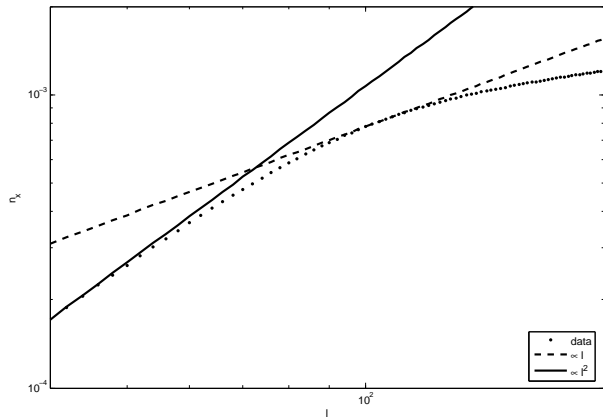


Abbildung 5.2: Abhängigkeit der maximalen Cluster-Dichte von  $l^y$

Während sich eine Proportionalität zu  $l^2$  für sehr schmale Terrassen erkennen lässt, zeigt sich im weiteren Verlauf, dass auch die Abschätzung  $n_x \propto l$  zu hoch ist. Solche Abschätzungen würde auch für wachsende Terrassenbreiten keine Annäherung an die maximale Cluster-Dichte bei ebener Oberfläche zulassen, was die Unterteilung in mehrere Phasen notwendig macht.

## 5.2 Simulations-Ergebnisse

Aus Abb. 4.7 lässt sich noch keine Aussagen über die Abhängigkeiten zwischen Cluster-Dichte und Terrassen-Breite ableiten. Eine bessere Vorstellung

vom Verhalten des Systems erhält man, wenn man durch die maximale Clusterdichte bei ebener Oberfläche teilt (Abb. 5.3). Hier wird deutlich, dass für große  $\frac{D}{R}$  der Quotient  $\frac{n_x}{n_0}$  wesentlich schneller abnimmt, ein exponentieller Kurvenverlauf scheint erkennbar.

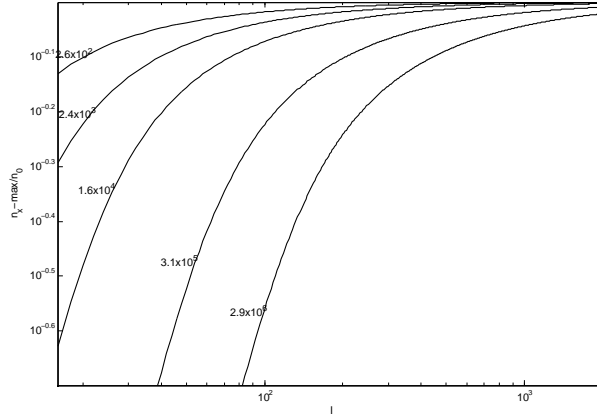


Abbildung 5.3: Maximale relative Cluster-Dichte in Abhängigkeit von der Terrassenbreite für verschiedene  $\frac{D}{R}$

### 5.3 Interpretation der Ergebnisse

Wie wir in 5.1 gezeigt haben, liefern bisherige Theorien mit  $n_x \propto l^y$  keine zufriedenstellende Übereinstimmung mit den Simulationsergebnissen. Wir gehen daher dem in Abb. 5.3 vermuteten exponentiellen Verlauf nach und wählen als Ansatz:

$$n_x \propto n_0 e^{g\left(\frac{D}{R}, l\right)} . \quad (5.4)$$

$n_0$  bezeichne die maximale Terrassenbreite auf ebener Oberfläche.

Um die gewünschten Nebenbedingungen zu erfüllen - vor allem für große Terrassenbreiten - muss die Funktion  $g$  für große  $l$  gegen 0 gehen und gegen  $-\infty$  für  $l$  gegen  $l_0$ . Die Terrassenbreite  $l_0$  sei also die, bei der sich keine Cluster mehr bilden. Naheliegender wäre es, diesen Wert auf 1 zu setzen, da sich auf Stufen, die breiter sind, als der Monomer-Durchmesser, gerade noch Cluster bilden können, die nicht an die Stufen angelagert sind.

Eine mögliche Lösung von  $g$  wäre:

$$n_x \propto n_0 e^{-\frac{f\left(\frac{D}{R}\right)}{l - l_0}} . \quad (5.5)$$

Um die These aus Glg. 5.5 zu prüfen, skizzieren wir den Verlauf von  $(\log \frac{n_x}{n_0})^{-1}$  für verschiedene  $\frac{D}{R}$ .

Abb. 5.4 zeigt den erwarteten Verlauf, für  $l$  im Bereich zwischen 1024 und 20. Für schmalere Terrassen zeigen sich Abweichungen, auf die wir in 5.3.2 eingehen werden.

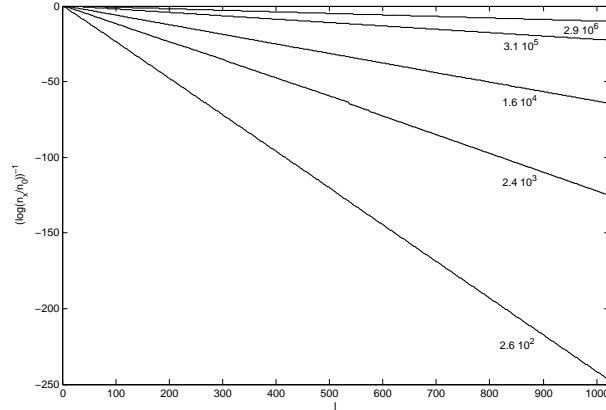


Abbildung 5.4: Verlauf von  $(\log \frac{n_x}{n_0})^{-1}$

### 5.3.1 Bestimmung von $f$

Zwar können wir für die in Abb. 5.4 dargestellten Geraden gut eine Steigung abschätzen, jedoch haben wir keinen durchgehenden Wertebereich zur Bestimmung von  $f$ . Wir betrachten daher Werte von  $f$  in Abhängigkeit von  $\frac{D}{R}$  für konstante  $l$ . Da die Steigung der Kurven (mit konstantem  $\frac{D}{R}$ ) stabil zu sein scheint, gehen wir davon aus, dass  $f$  nicht von  $l$  abhängt:

$$f = \log \frac{n_0}{n_x} \cdot (l - l_0) . \quad (5.6)$$

Wir wählen zunächst  $l = 512$ ,  $l = 128$  und  $l = 64$ , da in diesem Wertebereich  $\frac{1}{l} \approx \frac{1}{l-l_0}$ , wenn man davon ausgeht, dass  $l_0$  im Bereich zwischen 0 und 1 liegt. Im doppelt-logarithmischen Plot von  $f$  gegen  $\frac{D}{R}$  ergeben sich übereinstimmende Geraden (Abb 5.5). Daraus ergibt sich:

$$\log f = c + a \cdot \log \frac{D}{R} , \quad (5.7a)$$

und damit

$$f = d \left( \frac{D}{R} \right)^a . \quad (5.7b)$$

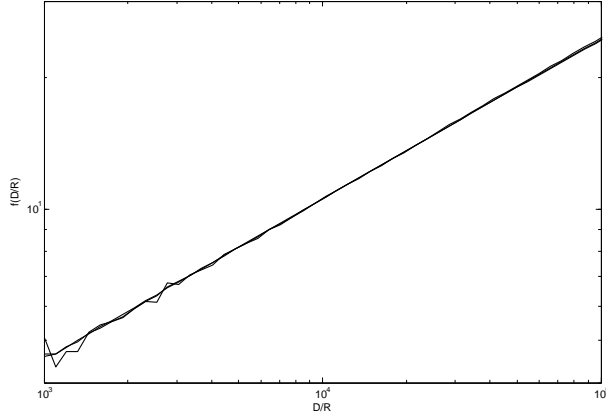


Abbildung 5.5: Darstellung von  $f$  im Verlauf von  $\frac{D}{R}$  für  $l$  zwischen 64 und 512

Aus angelegten Ausgleichsgeraden erhalten wir:  
 $c \approx -1$ ;  $a \approx 0.36$  und damit:

$$f \approx \frac{\left(\frac{D}{R}\right)^{0.36}}{e} . \quad (5.8)$$

Zur Kontrolle betrachten wir den Verlauf von  $\frac{n_x}{n_e}$  für  $l = 128$  in Abb. 5.6b.  $n_e$  bezeichne den nach Formel 5.5 errechneten Verlauf.

Ab  $\frac{D}{R} \approx 10^6$  zeigen sich deutliche Abweichungen zwischen  $n_x$  und  $n_e$ .

Dabei ist zu berücksichtigen, dass in diesem Bereich die maximale Cluster-Dichte erst bei  $\theta > \frac{1}{3}ML$  erreicht wird, so dass zusätzliche Effekte wie das Anwachsen an Stufen berechnet werden müssen. Außerdem ist für  $\frac{D}{R} > 10^5$  die maximale Cluster-Dichte ohnehin sehr gering, was großer Empfindlichkeit bei der Berechnung von  $\frac{n_x}{n_e}$  führt. Abb. 5.6a zeigt im beschränkten Bereich von  $\frac{D}{R}$  und für großes  $l$  bereits ein der Simulation recht ähnliches Verhalten von Glg. 5.5.

### 5.3.2 Wert von $l_0$ , schmale Terrassen

Für breite Terrassen lässt sich die maximale Cluster-Dichte gut annähern mit:

$$n_x \propto \left(\frac{D}{R}\right)^{-\frac{1}{3}} e^{-\frac{\left(\frac{D}{R}\right)^a}{e \cdot l}} . \quad (5.9)$$

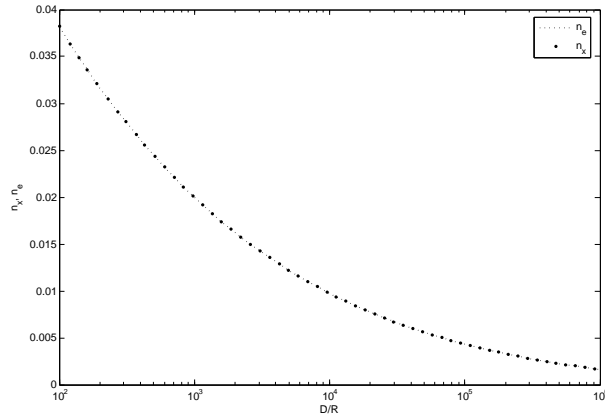


Abbildung 5.6a: Verlauf von  $n_x, n_e$  für  $l = 128$  und  $\frac{D}{R}$  zwischen  $10^2$  und  $10^6$

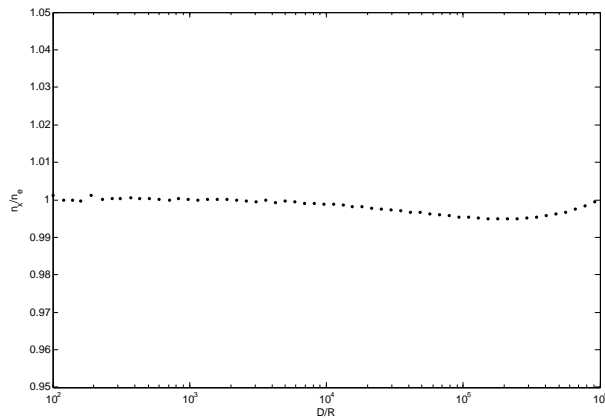


Abbildung 5.6b: Verlauf von  $\frac{n_x}{n_e}$

Mit bekanntem  $f$  betrachten wir jetzt die Werte für  $l_0$ . Aus Glg. 5.5 erhalten wir:

$$l - l_0 = \frac{f}{\log \frac{n_0}{n_x}} . \quad (5.10)$$

Da  $l_0$  im Idealfall eine Konstante sein sollte, widmen wir uns dem durch Glg. 5.8 gut beschriebenen Werte-Bereich und betrachten das Verhalten für kleinere  $l$ . Tragen wir für drei verschiedene  $\frac{D}{R}$  die nach Glg. 5.10 ermittelten Werte von  $l - l_0$  gegen  $l$  auf, ergeben sich beinahe identische Geraden mit einer Steigung von ca. 1 (Abb. 5.7).

Zwar lässt sich der Achsenabschnitt auf den Bereich um  $\pm 1$  eingrenzen, eine klare Bestimmung von  $l_0$  ist jedoch aufgrund unterschiedlicher Werte der drei Geraden nicht möglich. Da die Abweichungen für kleine  $l$  auch im Zusammenhang mit dem dort erst spät erreichten Maximum zusammenhängen können, vernachlässigen wir  $l_0$ .

Die leicht geringere Steigung der Geraden in Abb. 5.7 lässt darauf schließen,

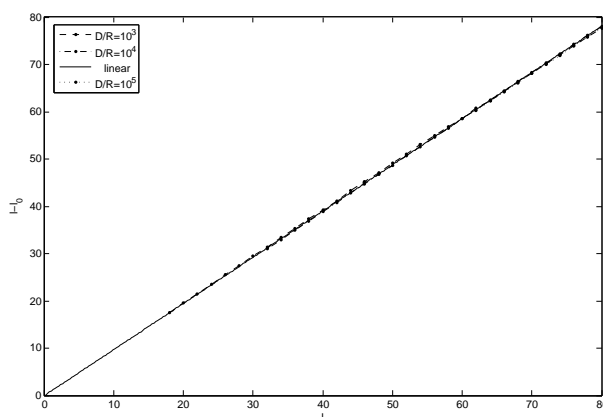


Abbildung 5.7: Bestimmung von  $l_0$  mittels Ausgleichgerade

dass wir den Wert von  $c$  in Glg. 5.7a leicht unterschätzt haben.

### 5.3.3 Ein Beispiel zur Probe

Wir beschreiben die maximale Clusterdichte auf vicinalen Oberflächen mit:

$$n_x = n_0 e^{-\frac{f(\frac{D}{R})}{l}} \propto \left(\frac{D}{R}\right)^{-\frac{1}{3}} e^{-\frac{f(\frac{D}{R})}{l}}. \quad (5.11)$$

Dabei entspricht  $f = \left(\frac{D}{R}\right)^a \cdot e^c$ .  $a$  knapp oberhalb von 0.36 und  $c$  knapp über  $-1$  lieferten die besten Resultate.

Wir wollen anhand eines Beispiels diese Darstellung noch einmal überprüfen. Dazu wählen wir  $\frac{D}{R} = 10^2$ , da für Werte in diesem Bereich auch kleine Stufenbreiten noch ein sehr gut erkennbares Maximum liefern. Deutlich ist nach wie vor ein leichtes Abknicken der Werte für kleine  $l$  zu erkennen (Abb. 5.8), was aber daran liegen kann, dass hier das Maximum erst für  $\theta \geq 0.3ML$  erreicht wird.

Die Gesamt-Abweichung liegt für  $l \geq 8$  bei ca. 0,5%, die Abweichung von  $n_x$  zu  $n_0$  liegt in diesem Bereich bei ca. 20%

Für  $l \geq 16$  und  $10^2 < \frac{D}{R} < 10^6$  wird das Verhalten der Simulationskurven

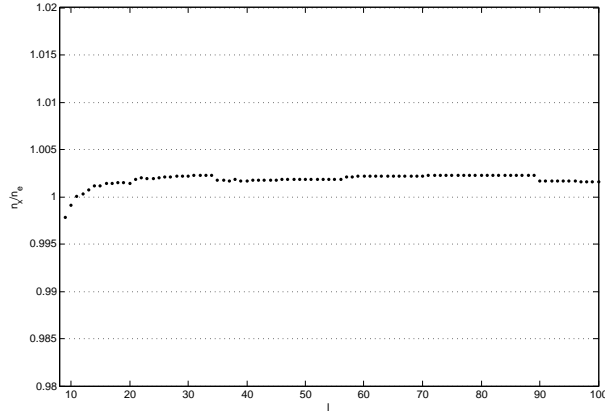


Abbildung 5.8: Verlauf von  $\frac{n_e}{n_x}$

durch Glg. 5.13 also gut angenähert.

Die Übertragbarkeit auf andere Werte (z.B. aus Experimenten oder Monte-Carlo-Simulationen) und andere Voraussetzungen (z.B.  $i > 1$ ) soll hier nicht thematisiert werden. Vor allem die Werte von  $a$  und  $c$ , die hier nur aus dem Vergleich mit Simulationskurven gewonnen wurden und nicht mathematisch hergeleitet sind, werden wahrscheinlich neu zu ermitteln sein.

## 5.4 weitere Simulationen

### 5.4.1 Korrekturterme - ein Versuch

Wir betrachten hier den Verlauf von Daten, die mit Glg. 3.19a und 3.19b ermittelt wurden, also mit Termen für die direkte Anlagerung.

Wir gehen im einfachsten Fall davon aus, dass sich ein ähnliches Verhalten für die Abhängigkeit von der Stufenbreite zeigen wird und beschränken daher den Korrektur-Term auf  $\frac{D}{R}$ . Zuerst widmen wir uns der Abhängigkeit zwischen maximaler Monomer-Dichte und  $\frac{D}{R}$  auf ebenen Oberflächen.

Geht man davon aus, dass  $n_0 \propto (\frac{D}{R})^{-\frac{1}{3}}$  und analog für die Terme mit direkter Anlagerung  $\tilde{n}_0 \propto (\frac{\tilde{D}}{R})^{-\frac{1}{3}}$ , ergibt sich:

$$\left(\frac{n_0}{\tilde{n}_0}\right)^3 \frac{D}{R} \propto \frac{\tilde{D}}{R} . \quad (5.12)$$

Abb. 5.9 zeigt, dass wir für den betrachteten Wertebereich in grober Näherung von einer Ursprungsgeraden ausgehen können, wodurch ein Ansatz der Form  $\frac{\tilde{D}}{R} \propto \frac{D}{R}$  sinnvoll scheint. Übertragen wir das auf Glg. 5.13 ergibt sich:

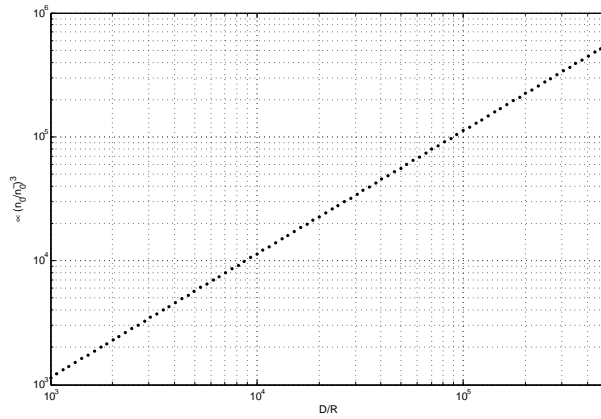


Abbildung 5.9: Vergleich von  $n_0$ ,  $\tilde{n}_0$

$$n_x = \tilde{n}_0 e^{-\frac{(p \cdot \frac{D}{R})^a}{l \cdot e}} \quad (5.13)$$

$p$  beschreibt dabei einen Proportionalitätsfaktor. Überprüft man diese grobe Näherung für ein Beispiel mit  $l = 64$ ,  $l = 128$  und  $l = 512$ , zeigen sich bei geringerer Stufenbreite bereits im Bereich von  $\frac{D}{R} \approx 10^5$  deutliche Abweichungen (Abb.5.10). Da sich hier die schon beschriebenen Probleme - vor

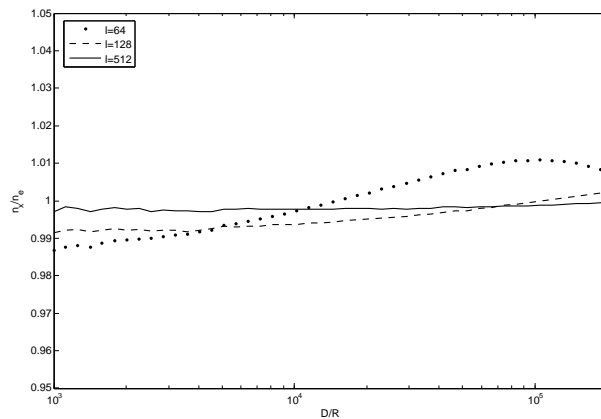


Abbildung 5.10: Annäherung für ein System mit direkter Anlagerung

allem das spät erreichte Maximum - bemerkbar machen, scheint ein Fehler im Prozent-Bereich auf ein Intervall von zwei Zehnerpotenzen jedoch annehmbar. Bestimmt man bei dieser Berechnung den Proportionalitätsfaktor

$p$  durch Minimierung der quadratischen Abweichungen, ergeben sich für alle betrachteten  $l$  Werte zwischen 0,94 und 0,98.

### 5.4.2 $n_x$ bei festem $\theta$

Im letzten Abschnitt setzen wir uns - statt mit der maximalen Cluster-Dichte - mit der Cluster-Dichte in einem sehr frühen Nukleations-Stadium auseinander. Wir wählen  $\theta = 0,05ML$  und betrachten die in Glg. 5.2 und 5.3 beschriebenen Proportionalitäten zu  $\theta$ ,  $l^3$ , bzw.  $l^4$  und zu  $\frac{R}{D}$ . Dazu betrachten wir zuerst den frühen Verlauf einiger Beispiel-Nukleationen (Abb. 5.11), dann für feste  $\frac{D}{R}$  die Abhängigkeit von  $l$  (Abb. 5.12) und schließlich die Proportionalität zu  $\frac{R}{D}$  (Abb. 5.13). Die Proportionalität zwischen  $n_x$  und  $\theta$  für die Anfänge der Nukleation lässt

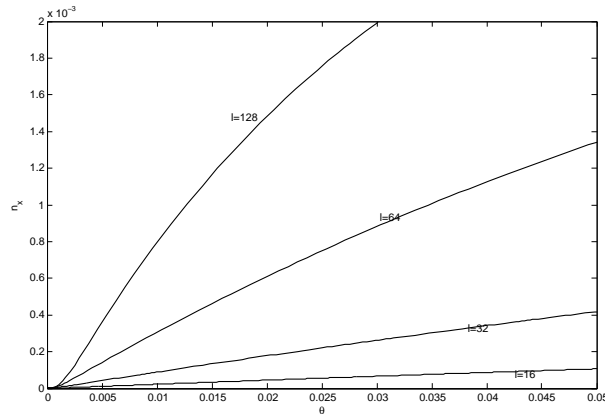


Abbildung 5.11: Frühe Nukleationsphasen für  $\frac{D}{R} = 10^5$  und verschiedene Clusterbreiten

sich in Abb. 5.11 gut erkennen. Allerdings wird für größere Terrassenbreiten bereits ab  $\theta \approx 0,02$  deutlich, dass das Wachstum der Cluster-Dichte schnell nachlässt.

Für die Abhängigkeit zwischen  $n_x$  und  $l$  zeigt das Modell ein von den Gleichungen abweichendes Verhalten. Wie in Abb. 5.12 dargestellt, ergibt sich auch für hohe  $\frac{D}{R}$  und geringes  $\theta$ :  $n_x \propto l^2$  und nicht wie vermutet  $l^3$ , bzw.  $l^4$ . Ein solcher Zusammenhang war auch in Abb. 5.2 für die maximale Cluster-Dichte eher zu vermuten.

In Abb. 5.13 ist deutlich zu erkennen, dass für schmale Stufen die Cluster-Dichte bei geringer Abdeckung proportional zu  $\frac{R}{D}$  wächst. Bei größeren  $l$  und kleinen  $\frac{D}{R}$  (also früh erreichter maximaler Cluster-Dichte) geht dieser Zusammenhang in Glg. 4.1 über.

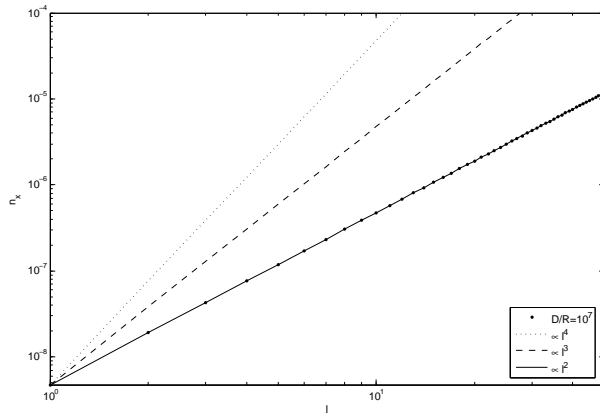


Abbildung 5.12: Proportionalität zwischen  $l^y$  und  $n_x$  für frühe Nukleationsstadien

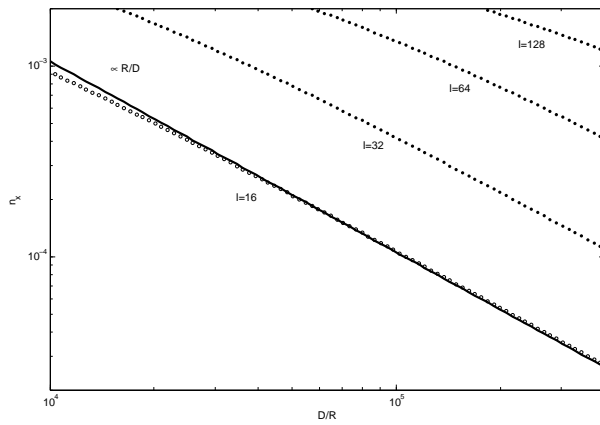


Abbildung 5.13: Proportionalität zwischen  $\frac{R}{D}$  und  $n_x$  für frühe Nukleationsstadien

Insgesamt lassen sich Proportionalitäten der Form  $n_x \propto a^b$  für geringe Abdeckungen gut darstellen.

## 6 Zusammenfassung / Conclusions

### Zusammenfassung

Bisherige Arbeiten zur Nukleation auf vicinalen Oberflächen basierten meist auf experimentell, oder per Monte-Carlo-Simulation errechneten Werten. In dieser Arbeit wurde das Thema mittels Ratengleichungen behandelt. Dieses ist weniger rechenaufwendig und ermöglicht eine wesentlich größere Bandbreite an ermittelten Daten. Allerdings sind für das Aufstellen der Gleichungen starke Vereinfachungen am System nötig.

Wir haben gezeigt, dass die maximale Cluster-Dichte auf vicinalen Oberflächen gut mit Hilfe der entsprechenden Werte für die ebene Oberfläche angenähert werden kann. Dieses hat den Vorteil, dass über Nukleation auf ebenen Oberflächen mehr bekannt ist, nicht zuletzt, da die mathematische Behandlung des Problems einfacher scheint.

Während für das einfache Modell die Annäherung nur über eine von  $\frac{D}{R}$  und  $l$  abhängige e-Funktion ausreichte, zeigte sich bereits für das Modell mit direkter Anlagerung, dass zusätzliche Korrektur-Terme nötig waren.

In wie weit die Ergebnisse auf komplexere Systeme, wie im Experiment vorhanden, übertragbar sind, ist schwer abzuschätzen. Dieses Modell beschreibt nur einen theoretischen Spezialfall, für eine allgemeine Beschreibung wären z.B. die Auswirkungen von beweglichen Clustern, Re-Evaporation und kritischen Keimen  $> 1$  zu bedenken.

Bekannte Näherungen der Form  $n_x \propto (\frac{D}{R})^{const}$  und  $n_x \propto l^{const}$  beschreiben zwar gut das Verhalten für sehr frühe Nukleationsphasen, versagen aber bei der Beschreibung der maximalen Cluster-Dichte.

## Conclusions

So far, papers on Nucleation on vicinal surfaces were mostly based on experimental data or Monte-Carlo simulations.

By using rate equations, we can calculate a wider range of data, without needing significantly more computing time. However, the model has to be simplified to find fitting equations.

We have shown, that the maximum Cluster-density on vicinal surfaces can be approximated by using the values for singular surfaces.

While the simple model can be approximated by an exponential-function of  $\frac{D}{R}$  and  $l$ , just adding direct impingement to the rate equations requires significant corrections in the approximation.

This model only deals with a very simple case of nucleation. It remains to be seen, whether the results can be used for more complex systems. Corrections for processes like reevaporation, the mobility of clusters and critical nucleus sizes of  $i > 1$  would certainly have to be made.

While well-known approximations like  $n_x \propto (\frac{D}{R})^{const}$  and  $n_x \propto l^{const}$  describe the early stages of nucleation rather well, they fail to describe the maximum cluster density.

## A Das Programm

Die in der Arbeit verwendeten Graphen stammen aus einem Matlab®-Programm, das sich aus zwei Teilen zusammensetzt:

Einer Benutzeroberfläche, die Daten einliest und einem Rechenprogramm. Hier soll nur eine kurze Beschreibung der Funktionsweise erfolgen, für detaillierte Informationen finden sich die kommentierten Programm-Texte in Anhang B.

### A.1 Die GUI

Die graphische Benutzer-Oberfläche (Abb. A.1) setzt sich im wesentlichen aus 5 Bausteinen zusammen:

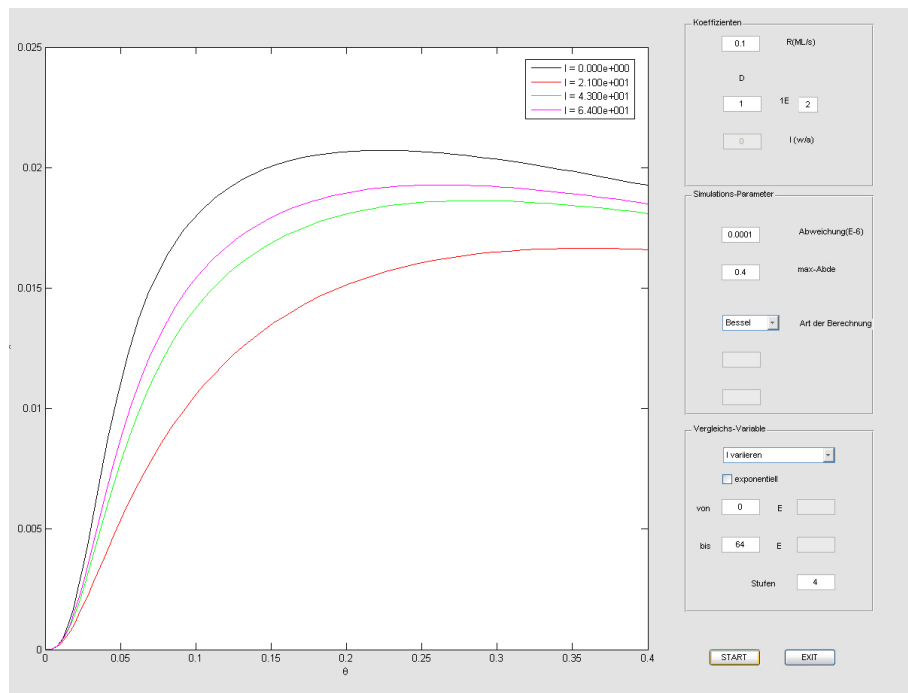


Abbildung A.1: Aufbau der Benutzeroberfläche

- Im Panel Koeffizienten finden sich Felder für die Eingabe der Depositionsrate, des Diffusionskoeffizienten und der Terrassenbreite. Wird eine Breite von 0 gewählt, simuliert das Programm eine ebene Oberfläche.
- Unter Simulationsparameter sind Kontroll-Elemente für die max. Abweichung, bzw. Schrittweite, die maximal-Bedeckung, sowie die Art

der  $\sigma$ -Berechnung untergebracht.

- Das Panel Vergleichsvariable bietet die Möglichkeit, für  $D$  oder  $l$  einen Werte-Bereich anzugeben. Das Programm erzeugt einen Koeffizienten-Vektor der angegebenen Länge, ggf. mit logarithmischer Einteilung. Diese Parameter werden dann seriell abgearbeitet.
- Im Grafik-Feld werden nach der Simulation die Ergebnisse für  $n_1$  und  $n_x$  dargestellt, bei Berechnung von 2 bis 6 Systemen nur  $n_x$ . Bei Berechnung von mehr als 6 Parametern wird nur der zuletzt erstellte Werte-Vektor dargestellt.
- Der Start-Button leitet eine Simulation, indem er alle Daten aus den Eingabe-Feldern einliest und überprüft. Wenn bei der Überprüfung der Parameter kein Fehler aufgetreten ist, ruft die GUI den Rechenkern entsprechend auf, stellt das zurückgegebene System dar und speichert das System mit den wichtigsten Variablen. Der Exit-Knopf beendet das Programm.

## A.2 Der Rechenkern

Der Rechenkern wird mit Werten für  $D$ ,  $l$ ,  $R$ , einer Maximal-Abweichung,  $\theta_{max}$  und Informationen zur  $\sigma$ -Berechnung aufgerufen.

Zunächst wird der Wert  $\frac{D}{R}$  berechnet, der für die weiteren Schritte immer wieder benötigt wird. Sollte eine Terrassen-Breite von 0 angegeben sein, wird die auf  $\infty$  gesetzt, so dass Terme der Form  $\frac{x}{l}$  verschwinden, was einer ebenen Oberfläche entspricht.

Die Querschnitts-Werte  $\sigma_1$  und  $\sigma_x$  werden mit entsprechenden Start-Werten initialisiert.

Dann wird die Funktion "nuk" mit entsprechenden Startwerten versehen mittels des Matlab©-ODE-Solvers (Details siehe Abschnitt A.4) gelöst. Innerhalb der nuk-Funktion befindet sich das eigentliche Gleichungssystem. Hier wird auch - so gewünscht - die Querschnitts-Berechnung vorgenommen.

## A.3 Bestimmung von $\sigma$

Die Bestimmung der Einfang-Querschnitte  $\sigma_1$  und  $\sigma_x$  geschieht je nach Wahl: Bei Wahl der konstanten  $\sigma$ s werden diese zugeordnet und entsprechend belassen.

Für den geometrischen Ansatz ergibt sich  $\sigma_1$  als 1,  $\sigma_x = \langle x \rangle^{\frac{1}{H}}$  wird in jedem Schritt neu berechnet. Die dazu benötigte durchschnittliche Cluster-Größe  $\langle x \rangle$  lässt sich nach Glg. 2.5 bestimmen.

Für die Berechnung mit Bessel-Funktionen verwendet der Rechenkern einen numerischen Trick. Um das Aufwendige Lösen der Gleichung 3.14 in jedem

Schritt zu umgehen, wird in Gleichung 3.15 das  $\sigma_x$  des letzten Zeitschritts verwendet.

Abbildung A.2 zeigt den Verlauf des Querschnitts bei nur 80 (!) Stützstellen im Vergleich zum korrekten  $\sigma_x$ . Nachdem die ersten drei Werte den Startwert korrigieren, zeigt sich kein nennenswerter Unterschied mehr zur Referenz-Kurve.

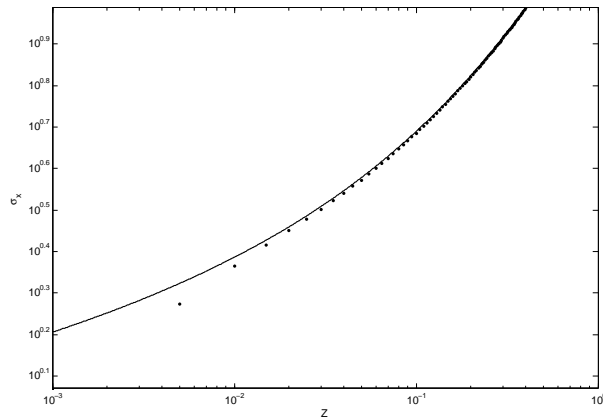


Abbildung A.2: Numerische Abschätzung von  $\sigma$

#### A.4 Der ODE-Solver

Die eigentliche Numerik wird vom Matlab©-ODE<sup>1</sup>-Solver übernommen. Der verwendete Solver ODE113 stützt sich auf das von Adams-Bashforth-Moulton entwickelten Mehrschritt-Verfahren. Es handelt sich hierbei um ein numerisches Verfahren vierter Ordnung [9]. Um Fehler bei der Verwendung des Solvers - vor allem bei der  $\sigma$ -Berechnung in der Funktion - auszuschließen, hier ein Vergleich:

Mittels eines simplen Euler-Verfahrens lässt sich - mit enormem Rechenaufwand - eine Vergleichskurve erzeugen. Da ein so einfaches Verfahren wenig systematische Fehler provoziert, legt ein Übereinstimmen der Verläufe nahe, dass die Solver-Ergebnisse den Verlauf richtig wiedergeben.

Ein direkter Vergleich von fünf unterschiedlichen Verläufen (auf ebener Oberfläche) lässt keinen Unterschied zwischen den Berechnungsarten erkennen (Abb. A.3a). Lediglich bei entsprechender Vergrößerung erkennt man, dass das Solver weniger Stützpunkte verwendet, die Kurve schneller Kanten zeigt (Abb. A.3b).

<sup>1</sup>ordinary differential equation

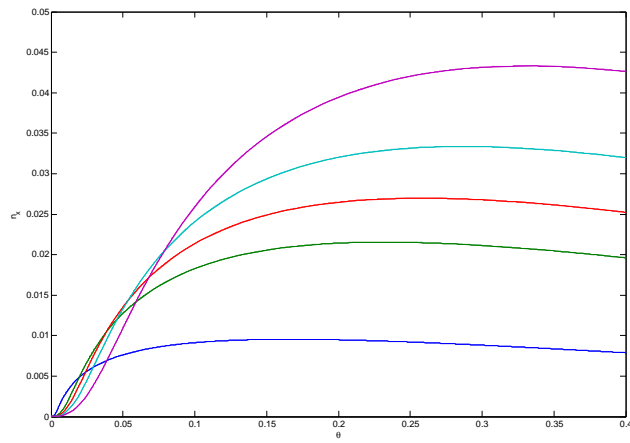


Abbildung A.3a: Vergleich zwischen numerischen Verfahren

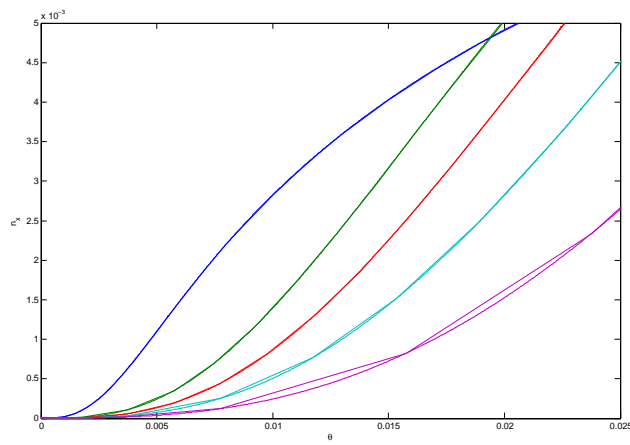


Abbildung A.3b: Vergrößerung eines Abschnittes aus Abb. A.3a

## B kommentierte Programm-Texte

### B.1 Die GUI

```
%GUI.m
%Graphische Benutzer-Oberfläche zur Bachelor-Arbeit
%"Nukleation auf vicinalen Oberflächen"

function varargout = GUI(varargin)
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                  'gui_Singleton',   gui_Singleton, ...
                  'gui_OpeningFcn', @GUI_OpeningFcn, ...
                  'gui_OutputFcn',  @GUI_OutputFcn, ...
                  'gui_LayoutFcn',  [], ...
                  'gui_Callback',    []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end

function GUI_OpeningFcn(hObject, eventdata, handles, varargin)
handles.output = hObject;
guidata(hObject, handles);

function varargout = GUI_OutputFcn(hObject, eventdata, handles)
varargout{1} = handles.output;

%Callback und Create Funktionen der diversen Eingabe-Felder
%funktions-frei
function edit1_Callback(hObject, eventdata, handles)
function edit1_CreateFcn(hObject, eventdata, handles)
if ispc
    set(hObject,'BackgroundColor','white');
else
    set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
end
function edit3_Callback(hObject, eventdata, handles)
function edit3_CreateFcn(hObject, eventdata, handles)
if ispc
    set(hObject,'BackgroundColor','white');
else
    set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
end
function edit4_Callback(hObject, eventdata, handles)
function edit4_CreateFcn(hObject, eventdata, handles)
if ispc
    set(hObject,'BackgroundColor','white');
```

```

else
    set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
end
function edit5_Callback(hObject, eventdata, handles)
function edit5_CreateFcn(hObject, eventdata, handles)
if ispc
    set(hObject,'BackgroundColor','white');
else
    set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
end
function edit11_Callback(hObject, eventdata, handles)
function edit11_CreateFcn(hObject, eventdata, handles)
if ispc
    set(hObject,'BackgroundColor','white');
else
    set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
end
function edit12_Callback(hObject, eventdata, handles)
function edit12_CreateFcn(hObject, eventdata, handles)
if ispc
    set(hObject,'BackgroundColor','white');
else
    set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
end
function edit13_Callback(hObject, eventdata, handles)
function edit13_CreateFcn(hObject, eventdata, handles)
if ispc
    set(hObject,'BackgroundColor','white');
else
    set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
end
function edit14_Callback(hObject, eventdata, handles)
function edit14_CreateFcn(hObject, eventdata, handles)
if ispc
    set(hObject,'BackgroundColor','white');
else
    set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
end
function edit16_Callback(hObject, eventdata, handles)
function edit16_CreateFcn(hObject, eventdata, handles)
if ispc
    set(hObject,'BackgroundColor','white');
else
    set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
end
function edit17_Callback(hObject, eventdata, handles)
function edit17_CreateFcn(hObject, eventdata, handles)
if ispc
    set(hObject,'BackgroundColor','white');
else
    set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
end
function edit22_Callback(hObject, eventdata, handles)
function edit22_CreateFcn(hObject, eventdata, handles)

```

```

if ispc
    set(hObject,'BackgroundColor','white');
else
    set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
end
function edit23_Callback(hObject, eventdata, handles)
function edit23_CreateFcn(hObject, eventdata, handles)
if ispc
    set(hObject,'BackgroundColor','white');
else
    set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
end
function edit26_Callback(hObject, eventdata, handles)
function edit26_CreateFcn(hObject, eventdata, handles)
if ispc
    set(hObject,'BackgroundColor','white');
else
    set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
end

function checkbox1_Callback(hObject, eventdata, handles)

%POPUP 1
%auswählen, welche Variable Parameter werden soll
%wenn 1: kein Laufparameter -> Param-Felder ghosten
%wenn 2: D Laufparameter
%wenn 3: l Laufparameter
function popupmenu1_Callback(hObject, eventdata, handles)
    if(get(handles.popupmenu1,'value')==1)
        set(handles.edit11,'enable','off');
        set(handles.edit12,'enable','off');
        set(handles.edit13,'enable','off');
        set(handles.edit14,'enable','off');
        set(handles.edit26,'enable','off');
        set(handles.checkbox1,'enable','off');
        set(handles.edit1,'enable','on');
        set(handles.edit4,'enable','on');
        set(handles.edit5,'enable','on');
        set(handles.edit3,'enable','on');
        set(handles.edit11,'String','');
        set(handles.edit12,'String','');
        set(handles.edit13,'String','');
        set(handles.edit14,'String','');
    end;
    if(get(handles.popupmenu1,'value')==2)
        set(handles.edit11,'enable','on');
        set(handles.edit12,'enable','on');
        set(handles.edit13,'enable','on');
        set(handles.edit14,'enable','on');
        set(handles.edit26,'enable','on');
        set(handles.checkbox1,'enable','on');
        set(handles.edit11,'String','1');
        set(handles.edit12,'String','2');
    end;

```

```

        set(handles.edit13,'String','1');
        set(handles.edit14,'String','5');
        set(handles.edit1,'enable','on');
        set(handles.edit4,'enable','off');
        set(handles.edit5,'enable','off');
        set(handles.edit3,'enable','on');
    end;
    if(get(handles.popupmenu1,'value')==3)
        set(handles.edit11,'enable','on');
        set(handles.edit12,'enable','off');
        set(handles.edit13,'enable','on');
        set(handles.edit14,'enable','off');
        set(handles.edit26,'enable','on');
        set(handles.checkbox1,'enable','on');
        set(handles.edit11,'String','0');
        set(handles.edit12,'String','');
        set(handles.edit13,'String','4096');
        set(handles.edit14,'String','');
        set(handles.edit1,'enable','on');
        set(handles.edit4,'enable','on');
        set(handles.edit5,'enable','on');
        set(handles.edit3,'enable','off');
    end;
function popupmenu1_CreateFcn(hObject, eventdata, handles)
if ispc
    set(hObject,'BackgroundColor','white');
else
    set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
end

%POPUP 2
%Wahl der Querschnittsberechnung:
%wenn 1:    konstante Sigmas
%wenn 2:    Sigma1 konstant, SigmaX nach geometrie-Formel
%wenn 3:    Bessel-Verfahren
%Funktionen nach betätigung des Popups:
%nicht benötigte Felder ghosten, Label-Texte aktualisieren,
%sinvolle Start-Werte einsetzen
function popupmenu3_Callback(hObject, eventdata, handles)
    if(get(handles.popupmenu3,'value')==1)
        set(handles.edit22,'enable','on');
        set(handles.edit23,'enable','on');
        set(handles.edit22,'String','3');
        set(handles.edit23,'String','8');
        set(handles.text22,'String','S1');
        set(handles.text23,'String','S3');
    end;
    if(get(handles.popupmenu3,'value')==2)
        set(handles.edit22,'enable','on');
        set(handles.edit23,'enable','off');
        set(handles.edit22,'String','1.7');
        set(handles.edit23,'String','');
        set(handles.text22,'String','H');
        set(handles.text23,'String','');
    end;

```

```

end;
if (get(handles.popupmenu3, 'value')==3)
    set(handles.edit22, 'enable', 'off');
    set(handles.edit23, 'enable', 'off');
    set(handles.edit22, 'String', '');
    set(handles.edit23, 'String', '');
    set(handles.text22, 'String', '');
    set(handles.text23, 'String', '');
end;
function popupmenu3_CreateFcn(hObject, eventdata, handles)
if ispc
    set(hObject, 'BackgroundColor', 'white');
else
    set(hObject, 'BackgroundColor', get(0, 'defaultUicontrolBackgroundColor'));
end

%Start-Button
%Simulations-Durchlauf starten
%Werte einlesen, Funktionen aufrufen, Graph bauen, Daten speichern
function pushbutton1_Callback(hObject, eventdata, handles)
%merken, ob alle eingelesenen Werte fehlerfrei
fehlerfrei = true;

%wenn Durchlauf:
%Lauf-Parameter einlesen, pruefen, ggf. Fehler melden, merken
if ~(get(handles.popupmenu1, 'value')==1)
    %Anzahl einlesen
    try
        zahl=str2num(get(handles.edit26, 'string'));
    catch
        error('Keine gueltige Parameter-Zahl!');
    end;
    %Parameter einlesen
    try
        grenze1=str2num(get(handles.edit11, 'string'));
        if (get(handles.popupmenu1, 'value')==2)
            grenze1=grenze1*10^(str2num(get(handles.edit12, 'string')));
        end;
    catch
        error('Ungueltige erste Grenze!');
        fehlerfrei=false;
    end;
    try
        grenze2=str2num(get(handles.edit13, 'string'));
        if (get(handles.popupmenu1, 'value')==2)
            grenze2=grenze2*10^str2num(get(handles.edit14, 'string'));
        end;
    catch
        error('Ungueltige zweite Grenze!');
        fehlerfrei=false;
    end;
    oben=max(grenze1, grenze2);
    unten=min(grenze1, grenze2);

```

```

%auf ungültige Werte prüfen
if(get(handles.popupmenu1,'value')==2)
    if (unten<1||unten==oben||oben>1E10)
        errordlg('Bitte Diffusionskoeffizienten zwischen 1 und 10^10 wählen!');
        fehlerfrei=false;
    end;
else
    if (unten<0||unten==oben||oben>10000)
        errordlg('Bitte Terrassenbreiten zwischen 0 und 10000 wählen!');
        fehlerfrei=false;
    end;
end;

%Parameter-Array bauen
if(get(handles.checkbox1,'value'))
    if(unten==0)
        Vars(1)=0;
        Vars(2:zahl)=logspace(log10(2),log10(oben),zahl-1);
    else
        Vars=logspace(log10(unten),log10(oben),zahl);
    end;
else
    Vars=linspace(unten,oben,zahl);
end;
Vars=round(Vars);
end;

%Werte einlesen, die Lauf-Params haetten werden koennen
%D, l einlesen, prüfen, ggf. Fehler melden, merken
if ~(get(handles.popupmenu1,'value')==2)
    try
        D=str2num(get(handles.edit4,'string'));
        D=D*10^(str2num(get(handles.edit5,'string')));
        if(D<1||D>1E10)
            errordlg('Bitte Diffusionskoeffizienten zwischen 1 und 10^10 wählen!');
            fehlerfrei = false;
        end;
    catch
        errordlg('Unguelziges D!');
        fehlerfrei = false;
    end
end
if ~(get(handles.popupmenu1,'value')==3)
    try
        l=str2num(get(handles.edit3,'string'));
        if((l<2||l>1E5)&&l~=0)
            errordlg('Bitte l-Wert zwischen 2 und kleiner 10000 wählen (0 für ebene Oberfläche!');
            fehlerfrei = false;
        end;
    catch
        errordlg('Unguelziges l!');
        fehlerfrei = false;
    end
end

```

```

end

%R, Simulations-Einstellungen einlesen,
%prüfen, ggf. Fehler melden, merken
try
    R=str2num(get(handles.edit1,'string'));
    if(R>10||R<=0)
        errordlg('Bitte R-Werte größer 0 und kleiner 10 wählen!');
        fehlerfrei = false;
    end;
catch
    errordlg('Unguelziges R!');
    fehlerfrei = false;
end
try
    Tmax=str2num(get(handles.edit17,'string'));
    if(Tmax>0.7||Tmax<=0)
        errordlg('Bitte maximal-Abdeckung zwischen 0 und 0.7 wählen!');
        fehlerfrei = false;
    end;
catch
    errordlg('Unguelzige Maximal-Abdeckung!');
    fehlerfrei = false;
end

try
    Dt=str2num(get(handles.edit16,'string'));
    if(Dt<=0)
        errordlg('Bitte maximal-Fehler größer 0 wählen!');
        fehlerfrei = false;
    end;
    if(Dt>=100)
        errordlg('Bitte maximal-Fehler kleiner 100 wählen!');
        fehlerfrei = false;
    end;
    Dt=Dt*1e-6;
catch
    errordlg('Unguelzige Schrittweite!');
    fehlerfrei = false;
end

%Art der Querschnittberechnung bestimmen
%Variablen einlesen
%je nach Art der Berechnung auf sinnvollen Wertebereich prüfen
%ggf. Fehler melden, merken
try
    Var1=str2num(get(handles.edit22,'string'));
    if(get(handles.popupmenu3,'value')==1&&(Var1<1||Var1>4))
        errordlg('Bitte S1 zwischen 1 und 4 wählen!');
        fehlerfrei = false;
    end;
    if(get(handles.popupmenu3,'value')==2&&(Var1<1||Var1>3))
        errordlg('Bitte H zwischen 1 und 3 wählen!');
    end;
end;

```

```

        fehlerfrei = false;
    end;
catch
    errordlg('Ungueltige Variablen in der Querschnittberechnung!');
    fehlerfrei = false;
end;

%nicht benötigte Variable initialisieren
%vermeidet Fehler beim Weiter-Schieben in den Rechenalgorithmus
if(get(handles.popupmenu3,'value')==2)
    Var2=0;
end;
if(get(handles.popupmenu3,'value')==3)
    Var1=0;
    Var2=0;
end;

%analog zweites Feld einlesen
if(get(handles.popupmenu3,'value')==1)
    try
        Var2=str2num(get(handles.edit23,'string'));
        if(Var2<5||Var2>10)
            errordlg('Bitte S2 zwischen 5 und 10 wählen!');
            fehlerfrei = false;
        end;
    catch
        errordlg('Ungueltige Variablen in der Querschnittberechnung!');
        fehlerfrei = false;
    end;
end;

%wenn beim Einlesen keine Fehler aufgetreten sind
if(fehlerfrei)
    %Plot freigeben
    hold off;

    %wenn Einzelberechnung: Durchführen!
    %System speichern
    if(get(handles.popupmenu1,'value')==1)
        [t,N,DR,l]=BeRechnung(D,R,l,Dt,Tmax,get(handles.popupmenu3,'value'),Var1,Var2);
        save('Sys','N','t');
    else
        %wenn parameter-Berechnung:
        %Parameter-Array durchlaufen, jeweils für entsprechenden Wert einsetzen
        %System berechnen, in Datei SysN speicher (N für Param-Nr.)
        %Label in Array einfügen, erleichtert später Legenden-Bau
        for(i=1:length(Vars))
            if(get(handles.popupmenu1,'value')==2)
                ['Beginne Durchlauf ', num2str(i), ' von ', num2str(zahl), '. D=', num2str(Vars(i))]
                [t,N,DR,l]=BeRechnung(Vars(i),R,l,Dt,Tmax,get(handles.popupmenu3,'value'),Var1,Var2);
                lab(i,:)=['D = ', num2str(Vars(i), '%10.3e\n')];
            end;
            if(get(handles.popupmenu1,'value')==3)
                ['Beginne Durchlauf ', num2str(i), ' von ', num2str(zahl), '. l=', num2str(Vars(i))]
            end;
        end;
    end;
end;

```

```

        [t,N,DR,l]=BeRechnung(D,R,Vars(i),Dt,Tmax,get(handles.popupmenu3,'value'),Var1,Var2);
        lab(i,:)=['l = ',num2str(Vars(i), '%10.3e\n')];
    end;
    name=['Sys',num2str(i)];
    save(name, 'N','t', 'DR', 'l');
end;
end;

%Abschluss:
%bunten Plot bauen
%Legende und Achsenbeschriftung einfügen
if(get(handles.popupmenu1,'value')~=1&&zahl<=6)
    bunt=['b';'k';'r';'g';'m';'c'];
    for(i=1:length(Vars))
        name=['Sys',num2str(i)];
        load(name);
        plot(t,N(:,2),bunt(mod(i,6)+1));
        hold on;
    end;
    ylabel('n_x');
    legend(lab);
else
    plot(t,N(:,1),t,N(:,2));
    ylabel('n_x, n_1');
    legend('n_1','n_x');
end;
xlabel('\theta');
end;

%Speicher fegen
clear;
hold off;

%Exit-Knopf
%beendet Programm
function pushbutton2_Callback(hObject, eventdata, handles)
delete(handles.figure1);

```

## B.2 Der Rechenalgorithmus

```

%BeRechnung.mat
%Berechnung von n1 und nX mit zugehöriger Zeitleiste (genauer: Theta-Leiste)

%Rückgabe: Theta-Leiste, System N
%Diffusions-Rate D
%Depositions-Rate R
%Terrassen-Breite l (0 für ebene Fläche)
%Abweichungs-Wert Delta
%maximal-Bedeckung Tmax
%Querschnittsberechnungs-Art art (1: konstant; 2: geometrisches VF, 3:
%Bessel-Verfahren)
%Querschnitts-Parameter var1, var2
function [t,N,DR,l] = BeRechnung(d,r,l,Delta,Tmax,art,var1,var2)

```

```

%globale Variablen in Funktion benötigt:
%einzel-Funktionen siehe unten
global DR L R f Art H SX S1 ART count;

%count-Variable: alle n Schritte Zwischenstand ausgeben
count=0;
%D/R: Diffusionskoeffizient D geteilt durch Depositionsrate R (siehe
%Ratengleichungen)
DR = d/r;
l;
%Terrassenbreite L:
%entweder l, oder unendlich
if(l==0)
    L=inf;
    f=1;
else
    f=(l-1)/l;
    L=l;
end;
R=r;
%Startwerte für die Querschnittsberechnung
ART=art;
if(art==1)
    S1=var1;
    SX=var2;
end;
if(art==2)
    H=var1;
    S1=1;
    SX=1;
end;
if(art==3)
    S1=1;
    SX=1;
end;

%Differentialgleichung mittels Matlab-Solver lösen
options=odeset('AbsTol',[Delta Delta Delta]);
%von 0 bis Tmax, Startwerte jeweils 0
[t,N]=ode113(@Nuk,[0,Tmax],[0,0,0],options);

%zu lösende Differentialgleichung
function dn = Nuk(t,n)
%globale Variablen siehe oben
global DR L R f Art H SX S1 ART count;
%Querschnitts-Berechnung entsprechend Auswahl

Z=t-n(1)-n(3);

if((ART==2)&&(n(2)~=0))
    SX=real((t-n(1)-n(3))/n(2))^(1/H);
end;

```

```

if(ART==3&&Z>0)
    SX=2*sqrt(pi*Z*SX)*besselk(1,sqrt(Z*SX/pi))/besselk(0,sqrt(Z*SX/pi));
end;
%Änderungen von n(1), n(2), n(3):
%n(1)=n1
%n(2)=nx
%n(3)=an Stufen angelagerte Monomere
dn      = zeros(2,1);
%Formeln mit oder ohne impingement
%dn(1)  = f-2*n(1)-Z-n(3)-DR*(2*S1*(n(1))^2+SX*n(1)*n(2)+n(1)/L);
%dn(2)  = DR*S1*n((1))^2+n(1)-2*n(2)*(1-dn(1));
%dn(3)  = DR*n(1)/L+R/L+n(3);
dn(1)   = 1-DR*(2*S1*(n(1))^2+SX*n(1)*n(2)+n(1)/L);
dn(2)   = DR*S1*n((1))^2-2*n(2)*(1-dn(1));
dn(3)   = DR*n(1)/L;
%ggf. Zwischenstand
%   if(mod(count,1000)==0)
%       t
%       SX
%   end;
%counter weiterzählen
count=count+1;

```

### B.3 Der Vergleichs-Rechenalgorithmus

```

%Vergleichsfunktion zum Prüfen der Matlab-ODE-Funktion
%Berechnung der analogen Gleichungen von Hand
%einfachstes 1-Schritt-Verfahren, sehr rechenaufwendig, bei großen
%Abweichungen
%Aber: Simpel genug um wenig systematische Fehler zu provozieren und für
%einen Vergleichslauf mit sehr geringer Schrittgröße durchaus zu ertragen
function [t,N,DR,l] = BeRechnung(d,r,l,Delta,Tmax,art,var1,var2)

%Vorberechnungen analog zur ODE-Variante
DR=d/r;
if(l==0)
    l=inf
end;

n1=0;
nx=0;
ns=0;

%Startwerte fuers S
if(art==1)
    S1=var1;
    SX=var2;
end;
if(art==2)
    H=var1;
    S1=1;
    SX=1;
end;

```

```

if(art==3)
    S1=1;
    SX=1;
end;

%ab hier:
%schrittweise abarbeiten der Gleichungen, aus dN wird Delta-N
%ebenso dt
pos=1;
for(i=0:Dt:Tmax)
    Dn1=1-2*n1-(i-n1-ns)-DR*(2*S1*n1^2+SX*n1*nx-n1/l);
    Dnx=DR*S1*n1^2+n1-2*nx*(1-Dn1);
    Dn1=Dn1*Dt;
    Dnx=Dnx*Dt;
    Dns=DR*n1/l*Dt;
    nx=nx+Dnx;
    n1=n1+Dn1;
    ns=ns+Dns;
    %bei großen Veränderungen Notbremse ziehen
    if(abs(Dnx)>0.0001||abs(Dn1)>0.0001)
        error('Schrittweite zu groß gewählt, Abbruch der Simulation');
        break;
    end;
    %System aufbauen
    Sys(pos, 1)=n1;
    Sys(pos, 2)=nx;
    Sys(pos, 3)=ns;
    %Position weitergehen
    pos=pos+1;
    %gggef Zwischenstand
    if(mod(pos,10000)==0)
        i
        (i-n1-ns)/nx
    end;
    %und S neu berechnen
    Z=t-n1-ns;
    if((art==2)&&(nx~=0))
        SX=real(Z/nx)^(1/H);
    end;
    if(ART==3&&Z>0)
        SX=2*sqrt(pi*Z*SX)*besselk(1,sqrt(Z*SX/pi))/besselk(0,sqrt(Z*SX/pi));
    end;
end;
%Zeitleiste muß von Hand gebaut werden, wird zur Kompatibilität mit GUI
%zurückgegeben.
t=0:Dt:Tmax;

```

## C weitere Abbildungen

### C.1 Auf ebener Oberfläche

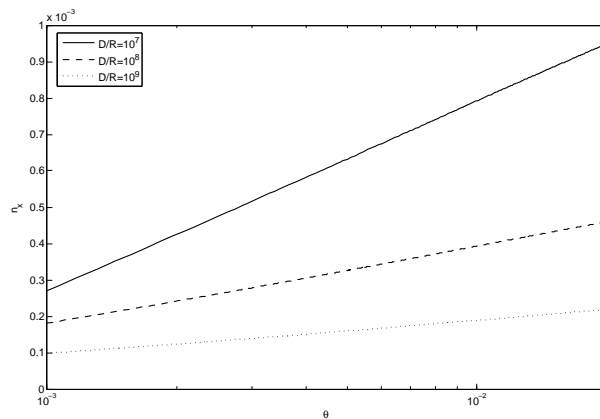


Abbildung C.1: logarithmisches Wachstum der Cluster-Dichte im Bereich kurz vor der Sättigungsphase (vgl. Abb.4.2, mittleres Segment)

### C.2 Auf vicinaler Oberfläche

Wir betrachten analog zu [4] in Abhängigkeit von  $l$  und  $\frac{D}{R}$ , wann eine kritische Cluster-Dichte  $n_x^c$  erreicht wird. Während für sehr hohe Clusterdich-

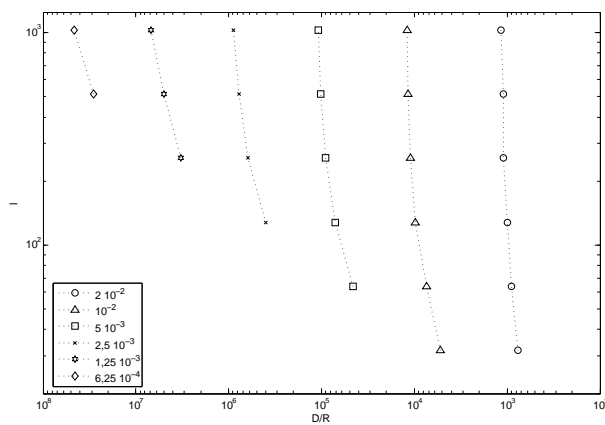


Abbildung C.2: Erreichen kritischer Clusterdichte für verschiedene  $n_x^c$

ten das Verhalten noch fast unabhängig von  $l$  scheint, lässt sich für geringer werdende Clusterdichte deutlich erkennen, dass für größere  $\frac{D}{R}$  auch breitere Cluster zum Erreichen einer vorgegebenen Cluster-Dichte notwendig. Analog zu den Überlegungen von Myers-Beaghton ( $l \approx 2(\frac{D}{R})^{\frac{1}{4}}$ ) ist auch diese Beziehung als Grenzbereich zwischen Terrassen- und Cluster-Nukleation als dominierenden Prozess anzusehen. Während im unten-links von der Verbindungslinie ein Großteil der Monomere sich an Stufen anlagern wird, überwiegt rechts oben die Bildung von Clustern.

Ein Problem des Modells ergibt sich dabei für kleine kritische Cluster-Dichten, da Verkleinerung von Terrassen durch Anlagerung, sowie das Heranwachsen von Clustern an Stufen in diesem Bereich eine wichtige Rolle spielen.

## Literatur

- [1] John Venables. *Introduction to Surface and Thin Film Processes*. Cambridge University Press, 2000.
- [2] Lorenz Nedelmann. *Nucleation, growth and strain relief in Cu hetero-epitaxy on Ni(100)*. PhD thesis, EPFL, 1996.
- [3] Jörg Rottler. Wachstumsprozesse an oberflächen und nukleation der zweiten lage. Master's thesis, Universität Konstanz, 1999.
- [4] Joachim Wollschläger. Kinetic phase diagram for terrace and step nucleation of caf/si(111).
- [5] G.S. Bales. Crossover scaling during submonolayer epitaxy on vicinal substrates. *Surface Science Letters*, 356:L439–L444, 1996.
- [6] J.A. Venables. Rate equation approaches to thin film nucleation kinetics. *Phil. Mag.*, 27:697–738, 1973.
- [7] J.G. Amar. Dynamic scaling of the island-size distribution and percolation in a model of submonolayer molecular-beam epitaxy. *Physical Review B*, 50, 1994.
- [8] Alberto Pimpinelli. Crossover scaling in thick film growth on vicinal surfaces. *International Journal of Modern Physics B*, 11 Nr.31:3647–3655, 1997.
- [9] Andy Voß. Mehrschrittverfahren. Vortrag im Proseminar Numerische Verfahren, Universität Hamburg, 2005.
- [10] [www.wikipedia.de](http://www.wikipedia.de).

## Abbildungsverzeichnis

2.1 Schematische Darstellung von atomaren Prozessen auf vicinaler Oberfläche. . . . .	6
3.1 Vergleich zwischen Uniform Depletion- und Lattice-Approximation, analog zu [6] . . . . .	12
4.1 Vergleich der Berechnungsmethoden von $\sigma_x$ für eine Abdeckung von $10^{-4}$ bis $10^{-1} ML$ . . . . .	16
4.2 Einteilung in Nukleations-Phasen . . . . .	16
4.3 Clusterdichte für verschiedene $\frac{D}{R}$ in Abhängigkeit von $\theta$ . . . . .	17

4.4	Maximale Cluster-Dichte in Abhängigkeit von $(\frac{D}{R})^{(-\frac{1}{3})}$ . . . . .	18
4.5	Durchschnittliche Clustergröße in Abhängigkeit von $\theta$ für $\frac{D}{R} = 10^5; 10^6; 10^7$ . . . . .	18
4.6	Darstellung der Clusterdichte bei $\frac{D}{R} = 10^5$ . . . . .	19
4.7	Maximale Cluster-Dichte in Abhängigkeit von der Terrassenbreite für verschiedene $\frac{D}{R}$ . . . . .	19
5.1	Vergleich zwischen Simulations-Ergebnissen und vorgegebenen Formeln für $l = 128$ . . . . .	21
5.2	Abhängigkeit der maximalen Cluster-Dichte von $l^y$ . . . . .	21
5.3	Maximale relative Cluster-Dichte in Abhängigkeit von der Terrassenbreite für verschiedene $\frac{D}{R}$ . . . . .	22
5.4	Verlauf von $(\log \frac{n_x}{n_0})^{-1}$ . . . . .	23
5.5	Darstellung von $f$ im Verlauf von $\frac{D}{R}$ für $l$ zwischen 64 und 512 . . . . .	24
5.6a	Verlauf von $n_x, n_e$ für $l = 128$ und $\frac{D}{R}$ zwischen $10^2$ und $10^6$ . . . . .	25
5.6b	Verlauf von $\frac{n_x}{n_e}$ . . . . .	25
5.7	Bestimmung von $l_0$ mittels Ausgleichgerade . . . . .	26
5.8	Verlauf von $\frac{n_e}{n_x}$ . . . . .	27
5.9	Vergleich von $n_0, \tilde{n}_0$ . . . . .	28
5.10	Annäherung für ein System mit direkter Anlagerung . . . . .	28
5.11	Frühe Nukleationsphasen für $\frac{D}{R} = 10^5$ und verschiedene Clusterbreiten . . . . .	29
5.12	Proportionalität zwischen $l^y$ und $n_x$ für frühe Nukleationsstadien . . . . .	30
5.13	Proportionalität zwischen $\frac{R}{D}$ und $n_x$ für frühe Nukleationsstadien . . . . .	30
A.1	Aufbau der Benutzeroberfläche . . . . .	33
A.2	Numerische Abschätzung von $\sigma$ . . . . .	35
A.3a	Vergleich zwischen numerischen Verfahren . . . . .	36
A.3b	Vergrößerung eines Abschnittes aus Abb. A.3a . . . . .	36
C.1	logarithmisches Wachstum der Cluster-Dichte im Bereich kurz vor der Sättigungsphase (vgl. Abb.4.2, mittleres Segment) . . . . .	49
C.2	Erreichen kritischer Clusterdichte für verschiedene $n_x^c$ . . . . .	49