Innovative Coating Solutions,
Place Saint-Pierre, 11, B-5380 Forville., Belgium
slu@incosol4u.com www.incosol4u.com

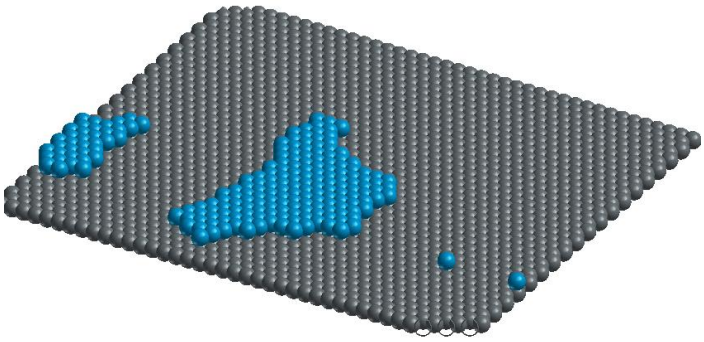
VIRTUAL COATER™

NASCAM™ (NanoSCAle Modeling)

Kinetic Monte Carlo code for the simulation of deposition, diffusion, nucleation and growth of a film on a substrate.

User's Manual for version 4.8

S. Lucas, P. Moskovkin, J. Müller



University of Namur (UNamur)
Namur Institute for Structured Materials (NISM)
Laboratoire d'Analyses par Réactions Nucléaires (LARN)
Innovative Coating Solutions (ICS)
https://www.incosol4u.com/nascam-general

Innovative Coating Solutions,
Place Saint-Pierre, 11, B-5380 Forville., Belgium
slu@incosol4u.com www.incosol4u.com

Table of Contents

1 Quick Start NASCAM Reference Card 4
2 Glossary 6
3 Introduction 9
4 Implementation 11
4.1 ALGORITHM 11
4.2 PHYSICS IMPLEMENTED IN THE CURRENT VERSION 13
4.3 ACTIVATION ENERGIES CALCULATION 18
4.3.1 NEB Method 18
4.3.2 Example: Cu/Cu(100) and Pt/Pt(100) 20
4.4 REACTIVE SPUTTER DEPOSITION 21
4.5 EVAPORATION 21
4.6 INCIDENT ENERGY DISTRIBUTION 23
4.7 INCIDENT ANGULAR DISTRIBUTION 25
4.8 MASKED DEPOSITION 27
4.9 SOURCE LINEAR MOVEMENT 28
4.10 SOURCE TILT AND ROTATION 29
4.10.1 EXAMPLES OF SOURCE MOVEMENTS: 30
4.11 CODE STRUCTURE 33
4.11.1 Initialization 33
4.11.2 Main part – kMC loop 33
4.11.3 Final part 34
5 Usage of the code 35
5.1 INPUT FILES 35
5.1.1 input.txt 35
5.1.2 substrate.xyz 41
5.1.3 Energy distribution data file 47
5.1.4 Angular distribution data file 47
5.2 OUTPUT FILES 49
5.2.1 coating.xyz and columns.xyz 49
5.2.2 film_growth.xyz 50
5.2.3 log_file.txt 50
5.2.4 stat_film_growth.csv 51
5.2.5 stat_film_structure.csv 52
5.2.6 final_composition.csv 54
5.2.7 stat_process.csv 54
5.2.8 island_distribution.csv 55
6 Execution of the code 56
7 JMOL plotting software 57
7.1 INTRODUCTION 57

Innovative Coating Solutions,
Place Saint-Pierre, 11, B-5380 Forville., Belgium
slu@incosol4u.com www.incosol4u.com

7.2 JMOL BASIC 58
8 Typical input data ranges 59
9 NASCAM version history 60
10 References 62
11 Citation 62
12 Annex 1: Another description of the input.txt file 64
13 Annex 2: Coordinate systems 67

Innovative Coating Solutions,
Place Saint-Pierre, 11, B-5380 Forville., Belgium
slu@incosol4u.com www.incosol4u.com

1 Quick Start NASCAM Reference Card

The package NASCAM of Virtual-Coater simulates atomistic deposition and metallic film growth on substrate. NASCAM can be run in two ways: using ASCII input files and running of the Nascam.exe or using the NASCAM_GUI - integrated user interface, which anyway calls the same Nascam.exe file. The user will find a manual of the NASCAM_GUI interface in a separate document.

1) Setup NASCAM's files:

input.txt: contains the current simulation parameters. The total number of lines is fixed and must be the one found in the manual. According to some command line switches, some lines are not read BUT have to be present. Table 1 contains a short description of the input.txt file.

Table 1. Short description of input.txt file content.

Names of fields in input.txt file	**Typical values for corresponding variables / *short description***
Simulation_options	12 1 0 */output a line in csv file every 12 deposited atoms, diffusion & island density computation switches (0-Not/1-Yes)*
Substrate_type	0 */0 = cubic; 1 = hexagonal*
Dimensions	3 3 10 2000 0 */ NXmax, NYmax, NZmax - system dimensions in lattice parameter units, NDep - no. of atoms to deposit, Tmax - annealing time(s)*
Deposition_rate,(ML/s)	0.05 */in Mono-Layer/Seconds (ML/s)*
Prefactor_correction	1.0 */1 for atoms, lower values for nanoparticles*
Masked_deposition	0 */0 = no mask, 1 = there is a mask; leave it at 0 to start with*
Pattern_name	LARN.txt */the name of the file containing the mask definition*
Source_linear_movement	0 0.0 */1st no. is a switch: 0 = not move, 1 = move; 2nd no. is a speed in lattice parameters/sec*
Ea_diff, ...(eV)	*Activation energies. If you have no clue, leave them to default values and go for RT deposition without diffusion (see Run NASCAM below)*
Specie:Metal_1	0.5 Co 63.546 1 0.029 eDist1.txt 1 80.0 0.0 1.0 aMDist1.txt */see below **
Specie:Metal_2	0.1 Co 63.546 2 20.0 eDist2.txt 1 0.0 0.0 1.0 aMDist2.txt */see below **
Specie:Reactive_1	1 O 32.0 1 0.029 eRDist1.txt 2 0.0 0.0 1.0 aRDist1.txt */see below **
Specie:Reactive_2	0.0 O 32.0 1 0.029 eRDist2.txt 2 0.0 0.0 1.0 aRDist2.txt */see below **
Specie:Gas	0.0 Ar 39.948 1 0.029 histoGas.txt 1 0.0 0.0 0.0 aGDist.txt */see below **
Specie:Substrate	Fe 28.085 */substrate nature & atomic mass*
Temp,(eV)	0.035 */deposition temperature: 75 °C = 0.03 eV*
Save_data	1 10000 */1st no. is a switch: 0 = do not save, 1 = save intermediate results every 10000 deposited atoms*
Surface_binding_energy _of_the_substrate,(eV)	1.0 */e.g. TRIM values (heat of sublimation): Al = 3.36; Si = 4.7; Ti = 4.89; Cr = 4.12; Cu = 3.52 ...*
Surface_binding_energy _of_the_film,(eV)	3.0
Sputtering_Thresh/Yield	0.0 0.0 */currently not in use/*

w_rot(1/s);init_tilt(deg); A_osc(deg);w_osc(1/s)	0.0 0.0 0.0 0.0 */substrate rotation speed, initial tilt angle, oscillation amplitude and oscillation frequency*
Forced_deposition	1.0E15 */a limit for the number of diffusion events that can happen between two successive deposition events*
Stoichiometry index	1.5 /*Stoichiometry index (SI) is the maximum relative amount of the reactive specie in the compound. Examples: SI = 1 for ZnO, SI = 1.5 for Al2O3, SI = 2 for TiO2*/
Bonding	5.0 /*currently not in use*/
* Specie:Metal_1	1.0 Co 63.546 1 0.029 eDist.txt 1 80.0 0.0 1.0 aMDist.txt
	fraction of mobile particles in the total flow: 1.0 *nature and atomic mass of the depositing particle: Co 63.546* *energy switch: 1* *energy mean value: 0.029* *energy distribution filename: eDist.txt* *angle switch: 1* *theta angle θ_0: 80.0°* *phi angle φ_0: 0.0°* *delta theta $\Delta\theta$: 1.0°* *angle distribution filename: aMDist.txt* *energy switch (metal & gas):* *0 ⇒ read energy distribution from a data file (e.g. eDist.txt)* *1 ⇒ use the given energy mean value E_0 (e.g. E_0 = 0.029eV)* *2 ⇒ compute energy using the analytical function $f = \exp(-E/E_0) / E_0$.* *angle switch:* *0 ⇒ read angular distribution from a data file (e.g. aMDist.txt)* *1 ⇒ use the given values for θ_0, φ_0 and $\Delta\theta$* *1 ⇒ uniform angular distribution (used for chamber gas, for example)*

substrate.xyz: contains a geometrical and chemical description of the substrate. If the *substrate.xyz* file is not present in the current folder or if it is but empty, the substrate is considered flat and having the physical size and chemical composition as defined in the *input.txt* file.

2) Run NASCAM:

- Double click on Nascam.exe.
- Use command line switches in the Windows console: *Nascam.exe switches, where possible switches are:*
  - *-n filename* defines the name for an optional input file. If the switch is used this file will be used instead of *input.txt* file.
  - *-ver* checks the version of the code without doing any simulations.

3) To analyse results

Look at the following files:

- *log_file.txt:* Deposition as it occurred
- *stat_film_growth.csv:* Evaporated atoms, Sputtered atoms, Recoil, Detached atoms, Detrapped atoms, Nbre of jumps, Gas atoms in layers, ...
- *stat_film_structure.csv:* Metal atoms, Gas atoms, Defects, Free atoms, Dimers, Islands, Film roughness, Substrate coverage, Density, Mean diffusion path

- *stat_process.csv:* Incident atom, Incident angle, Incident energy, I-J-K, Hitted specie, Transferred energy
- *final_composition.csv:* Chemical depth profile of the film

4) Plot in 3D the following files: *substrate.xyz* (substrate)*; film_growth.xyz* (intermediate results)*; coating.xyz and deposited_layer.xyz* (final results) have the right format to be plotted with JMOL software or any other software accepting XYZ file format.

2 Glossary

Adatom:	any deposited (or pre-existing) atom on the substrate surface or on the surface of deposited film that can move according to one of the possible mechanisms. Sometimes we call it mobile particle, too.
Atom:	a nucleus surrounded by electrons. It is considered as a hard sphere. Sometimes we call it particle, too.
Characteristic time:	interval of time between two successive deposition events (corresponding to real or equivalent particles).
Defect:	occupied lattice position modifying the physical properties of the structure in its neighbourhood and acting as a trap for a diffusing particle. Only particles belonging to the substrate can act as defects.
Diffusing particle:	generally a mobile particle having no nearest neighbour above and an available (empty) position as final position for at least a diffusion event.
Element name:	periodic table element name used as a label for a particle. In NASCAM, Xx stands for defect.
Equivalent particle:	a non-real particle used for time counting during an annealing process.
External/Internal coordination system	two different coordination systems are used by NASCAM: an external one when read and write all input/output system structure information (.xyz files), and an internal one used for computational purposes only (see Annex 2).
Event:	any movement (diffusion) of an atom from one location (lattice site) to another one; the deposition and evaporation are also considered as events: the evaporation is a thermal activated process (like diffusion), while the deposition not.
Free atom:	an atom that has no nearest neighbours at the same height as its own height (no lateral nearest neighbours) and no nearest neighbour above. There are free atoms on the substrate and on the deposited film.

Island: a well-defined 2d or 3d structure, the result of the nucleation process and, further, growth process by the attachment of new particles. For NASCAM two particles being nearest neighbours each other is considered the minimal configuration defining an island – the dimer of particles (island size starts at 2 atoms).

Isolated atom: a particle that has no other particle as a nearest neighbour. For a given lattice site, NASCAM distinguishes between its lateral nearest neighbours sites, those surrounding this site in a plane parallel to the bottom surface of the simulation box, and the nearest neighbour sites belonging to the layer above or below. These two last nearest neighbour sites are called above/below (sometimes top/bottom) nearest neighbour sites. An isolated atom (particle) is surrounded by six empty nearest neighbour sites for the cubic symmetry and eight for the hexagonal one.

Lattice parameter unit: the distance between two nearest neighbours is considered as a dimensionless distance unit (equal to 1). All distances are given in real numbers multiplying this lattice parameter unit. To have the real x, y and z dimensions of the system, the system dimensions given in lattice parameter units have to be multiplied by the lattice parameter values (usually in Å or nm) corresponding to the system coordination axis.

Mechanism: physical process which changes the morphology of the substrate surface, such as diffusion, evaporation, etc. One mechanism includes or is the result of several events.

Nearest neighbours (nn) & Next nearest Neighbours (nnn): for a given lattice symmetry, computing all possible distances between two lattice sites and ordering them according to their values, the distance minimum value corresponds to the distance between a given site and all its closest sites. All these sites are the nearest, or the first order, neighbour sites for the given site. They are six for the cubic symmetry and eight for the defined hexagonal symmetry. The next distance value corresponds to the distance between a given site and each of its next nearest, or second order, neighbour sites. They are twelve for the cubic symmetry and also twelve for the defined hexagonal symmetry.

Particle: an atom occupying a lattice site (substrate, gas, defect or adatom).

Sample: a snapshot of the growing structure including the substrate and deposited atoms.

Substrate: a 3d structure already existing inside the simulation box before starting the current simulation, and being different in its nature from the depositing particle (or adatoms). Under certain conditions, pre-existing adatoms can be considered as substrate particles (they cannot diffuse) keeping their chemical identity. It may contain defects, may be flat, be corrugated, etc.

Innovative Coating Solutions,
Place Saint-Pierre, 11, B-5380 Forville., Belgium
slu@incosol4u.com www.incosol4u.com

3 Introduction

The present code is developed to simulate at the atomic scale the time evolution of a system under deposition or/and annealing conditions using the kinetic Monte Carlo (kMC) method [1-5]. Because kMC method does not take into account the vibrational movement of atoms it can be used to study the evolution of a system for longer periods of time (similar to a real experiment - seconds to minutes) than the Molecular Dynamics method (~10^{-9}s). Depending on the number of atoms in the system, along with some other parameters such as temperature and atomic properties, the empirical (as opposed to simulation) time might be even hundreds to thousands of seconds.

There are several approaches in implementing the kMC method. Creating and using a complete kMC table of events is the most accurate approach. It is the most time-consuming method, too, as it is necessary to calculate the rates for a huge number of events. In the case of a kMC bond-counting approach, one calculates the event rates taking into account the number of nearest neighbours. In a specific kMC event approach, only a predefined set of events is allowed to occur, and the rates of these events are used as an input. This is the kMC approach used by NASCAM.

Although the last approach is not as accurate as using a complete table of events, it allows for simulating the general behaviour of a given system and requires less computational time, making it possible to simulate the evolution of a system with a larger number of particles. This is important especially for modelling a system at elevated temperatures.

The total number of steps required to simulate a system evolution during a given period of time can be derived as follows. Suppose N physical mechanisms thermally activated (e.g. free particle diffusion, detachment from a step edge, evaporation etc.). We call an event any movement that can be associated with the physical mechanisms. The events can happen in given directions (e.g. diffusion to the right or left). We call ν_i the number of possible directions for a given mechanism *i*.

Elementary event rate for a given event *i* obeys an Arrhenius law:

$$w_i = w_0 \exp(-E_i / k_B T), \qquad (1)$$

where w_0 is the attempt frequency (it can be estimated as $w_0 = 2 k_B T/h$), and E_i is the activation energy for the event *i*.

The time step between two events is defined as

$$\Delta t = \left(\sum_{i=1}^{N} \nu_i w_i \right)^{-1}, \qquad (2)$$

where the summation is done over all possible event mechanisms, *N*, in the system. If for a given particle *j* we have $\nu_i(j)$ possible diffusion directions corresponding to the event mechanism *i*, the previous relation can be re-written as:

$$\Delta t = \left(\sum_{i=1}^{N} \left(\sum_{j=1}^{N_{at}(i)} v_i(j) \right) w_i \right)^{-1}, \qquad (3)$$

where $N_{at}(i)$ is the total number of particles that can participate in the event with index i.

If we consider that the diffusion can happen only between two nearest neighbour positions and $v_i(j)$ can have values starting from zero to the nearest neighbour positions number, then the second summation in (3) can be extended over the total number of particles in the system, N_{at_tot}:

$$\Delta t = \left(\sum_{i=1}^{N} \left(\sum_{j=1}^{N_{at\_tot}} v_i(j) \right) w_i \right)^{-1}, \qquad (4)$$

From equations 1 and 3, one can estimate the total number of steps required to model a system evolution during a given period of time t_{tot} as $N_{steps} = t_{tot}/\Delta t$:

$$N_{steps} = t_{tot} \left(\sum_{i=1}^{N} \left(\sum_{j=1}^{N_{at\_tot}} v_i(j) \right) w_0 \exp(-E_i / k_B T) \right). \qquad (5)$$

If the temperature is high enough to neglect the contributions of all events excepting the one corresponding to the lowest energy barrier (the most common event), s, then equation 5 becomes:

$$N_{steps} = t_{tot} v_s w_0 \exp(-E_s / k_B T) = t_{tot} \left(\sum_{j=1}^{N_{at}(s)} v_s(j) \right) w_0 \exp(-E_s / k_B T). \qquad (6)$$

Therefore, the total computation time is mostly related to the event having the lowest energy barrier. For example, a temperature rise of 50°C leads to an increase of the required kMC steps by an order of magnitude. Nevertheless, number of particles as large as 10^4 and time evolution of the system within 10^2 – 10^3 s are still acceptable for NASCAM even for temperatures as high as 1000 °K.

Another characteristic of such kMC approach when used for high temperature simulations is that, for two mechanisms with close activation energies, the same activation energy can be used for both because the event rates are about the same ($w_1/w_2 = exp(-\Delta E_{12}/k_B T)$) as the temperature rises.

Lastly, and connected with the previous observation, the chosen kMC method gives the possibility to perform high temperature simulations using parameters that are not known precisely; this is because errors in determining the values for barrier energies do not lead to large changes in event rates.

Innovative Coating Solutions,
Place Saint-Pierre, 11, B-5380 Forville., Belgium
slu@incosol4u.com www.incosol4u.com

4 Implementation

4.1 ALGORITHM

The NASCAM algorithm is given in the figure 1.

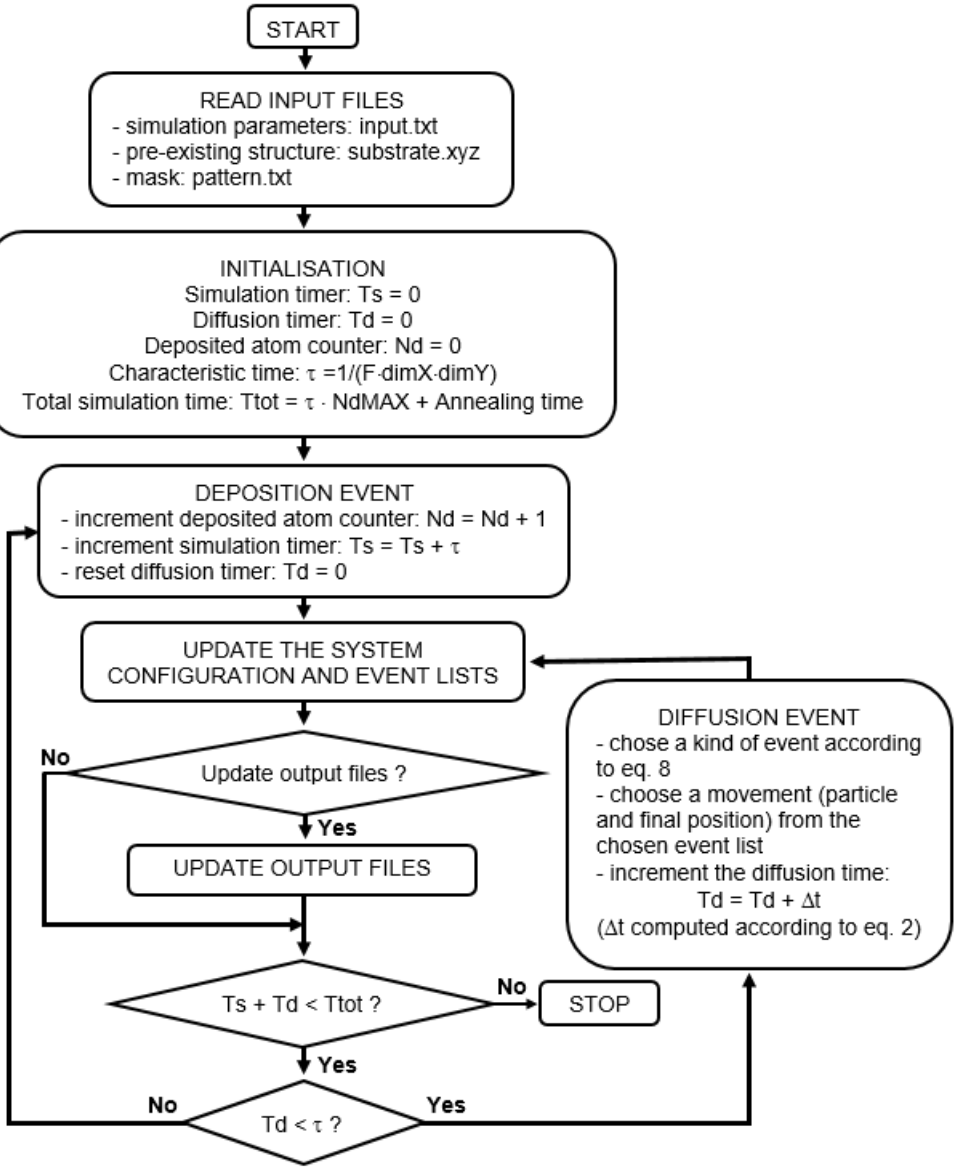


Figure 1. Algorithm of NASCAM code.

The atoms are deposited on the substrate at random positions at equal time intervals. The time interval, the characteristic time of the system - τ, is determined by the deposition rate and dimensions of the system substrate:

$$\tau = \frac{1}{F \cdot \dim X \cdot \dim Y}, \qquad (7)$$

Innovative Coating Solutions,
Place Saint-Pierre, 11, B-5380 Forville., Belgium
slu@incosol4u.com www.incosol4u.com

where F is the deposition rate (in ML/s), *dimX* is the system dimension on x and *dimY* is the system dimension on y (both in lattice parameter units), all three parameters given in the *input.txt* file.

In the current model only diffusion or evaporation events can take place between two successive deposition events. Once deposited, a particle and the particles in its neighbourhood are checked for all possible movements and placed, if necessary, into the corresponding atom-event lists. There are as many event lists as many possible thermal activated events are taken into account. Currently, NASCAM takes into account 9 such events. Their list and supplementary explanations are given in the next chapter. For the sake of completeness, it has to be mentioned that an adatom can appear into a certain event list more than once. Let us consider, for example, a square symmetry, and an adatom sitting on a terrace, having no lateral nearest neighbours and far from any steps. Let us also consider that the diffusion is possible only between two nearest neighbour positions. Under such suppositions our adatom appears into the isolated atom diffusion list four times, once for each possible diffusion direction. In fact, after each event (deposition/diffusion/evaporation), for each defined possible thermal activated event, if necessary, the corresponding list of atoms is updated (created if we are in the beginning of a new simulation).

The system time evolution is determined by the probabilities of the events that may occur during the current simulation. The procedure for selecting an individual event corresponding to a certain simulation step is divided into two sub-steps.

First, a type of event is chosen according to its occurrence probability p_i. This is given by the ratio of the rate corresponding to the event i (i. e. $N_i w_i$,) and the total rate computed as the sum over all event rates:

$$p_i = N_i w_i / \sum_i N_i w_i \ , \qquad (8)$$

where N_i is the number of atoms in the i-event list and w_i is the rate of such elementary event. Second, an atom is chosen randomly among the N_i atoms belonging to the event list i. At each simulation step, a time increment is calculated according to equation 2 (or, equivalent, eq. 3). When the sum of the time increments in the sequence of individual events becomes larger than the characteristic time of the system, a new atom is deposited.

4.2 PHYSICS IMPLEMENTED IN THE CURRENT VERSION

In the current model, 9 thermal activated events are taken into account. We give below a classification of these events, classification based on the local configuration of both the initial and final positions for a diffusing particle. For each kind of event a snapshot of its occurring conditions is also given, for the sake of clarity, in a 2D representation. The symbols used in these snapshots have the following meaning:


– a lattice site having an occupancy that is not relevant for the event under consideration

– a nearest neighbour site of the site occupied by the diffusing particle: according to the implemented diffusion model if such site is occupied by a mobile particle (same nature as the diffusing particle) that has only one nearest neighbour (this nearest neighbour is, obviously, the diffusing particle itself), the diffusion can be forbidden if this second mobile particle becomes after diffusion an isolated particle – a particle without nearest neighbours (6 for cubic symmetry and 8 for hexagonal symmetry)


– an empty lattice site


– a lattice site occupied by a particle (substrate, gas or mobile particle)


– a lattice site occupied by a defect (substrate particle)


– a diffusing/evaporating particle

– a possible event (diffusion/evaporation) direction


– a forbidden event (diffusion/evaporation) direction

In all figures from 2 to 7 the diffusing/evaporating particles (in green) are considered to be at a height $z = n$ while the occupied lattice sites (in blue) are at a height $z = n - 1$.

If the initial position has a defect as a nearest (Fig. 2a and 2b) or next nearest neighbour (Fig. 2c, 2d and 2e) in the layer below, only one kind of event can occur: the detrapment (from a substrate defect) – a jump to a new position belonging to the same layer or to the layer below, activation energy being Ea_detrap.

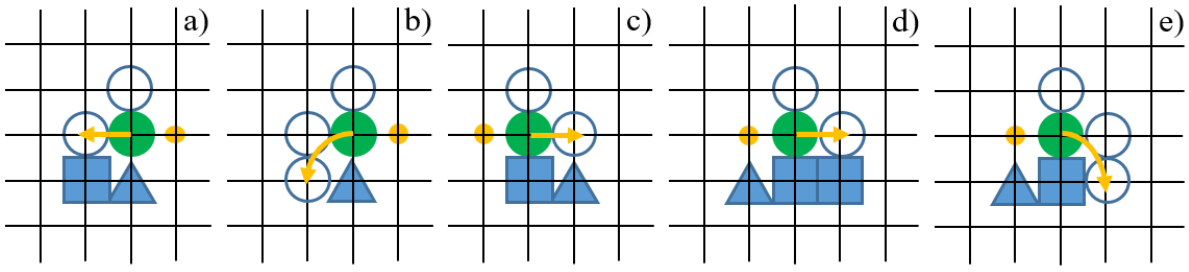


Figure 2. Detrapment events (see details in the text).

As long as the initial position has no defect as a nearest or next nearest neighbour in the layer below, the possible events can be classified into three categories:

a) diffusion events with both, the initial and final position having the same z-coordinate (in-layer diffusion). The initial and final position are nearest neighbours each other. A supplementary condition is that the below nearest neighbour site of the final position must be always occupied by another particle. According to the local configuration there are 4 kinds of in-layer diffusion events:

   a1) free mobile particle diffusion, activation energy being Ea_diff

   a2) jump to a new position having the same or larger number of lateral nearest neighbours than the initial position of the diffusing particle has, activation energy being Ea_nn_inc

   a3) jump to a new position having a lower number of lateral nearest neighbours than the initial position of the diffusing particle has, activation energy being Ea_nn_dec

   a4) detachment of a particle from a step edge bordering an island or a terrace, the particle having in its final position no lateral nearest neighbours, activation energy being Ea_detach

   For all these events, the corresponding "main" local configuration is given in fig. 3. Of course, each event from a1) to a4) has its own specific configuration in what concerning the lateral nearest neighbours of the initial and final site of the diffusing particle. There are two other possible movements that, even if the diffusion particle moves along the z axis of the coordination system, can be considered according to this classification as in-layer diffusions. A detailed explanation and the corresponding local configurations are given below.

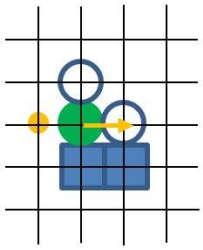


Figure 3. In-layer diffusion.

b) diffusion events with the initial and final position having different z-coordinates (layer-to-layer diffusion). The initial and final position are next nearest neighbours each other. According to the local configuration there are 2 kinds of the layer-to-layer diffusion events:

   b1) jump up - diffusion event having Ea_up as activation energy (Fig. 4a)

   b2) jump down - diffusion event having Ea_down as activation energy (Fig. 4b and 4c)

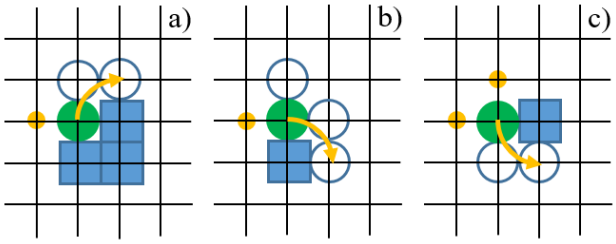


Figure 4. Jump up (a) and jump down (b, c) diffusion events.

c) evaporation of a free mobile particle happens when all lattice sites belonging to its trajectory to the vacuum, and all their lateral nearest neighbours sites, are empty sites (Fig. 5). The trajectory is always along the z axis of the coordination system and the vacuum limit is the upper surface of the simulation box. According to the nature of the below nearest neighbour of the evaporating particle there are two kinds of evaporation events:

   c1) evaporation from substrate, characterized by Ea_sub_evap activation energy, when the below nearest neighbour of the evaporating particle is a substrate particle

   c2) evaporation from layer, characterized by Ea_lay_evap activation energy, when the below nearest neighbour of the evaporating particle is a gas or another mobile particle

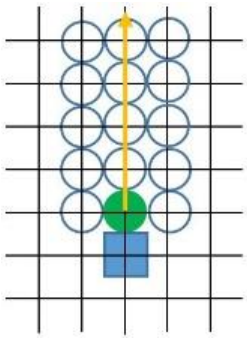


Figure 5. Evaporation event.

There are two “special” kinds of diffusion events that can be classified in both in-layer and layer-to-layer diffusion categories: the “moving-up” (Fig. 6a) & “moving-down” (Fig. 6b) diffusion events. They correspond to a movement along z axis. Such event requires the existence of at least one occupied lateral nearest neighbour site of the initial position of the diffusing particle. This occupied lateral neighbour site must have at its turn, according to the movement orientation (up or down) an occupied above or below nearest neighbour site.

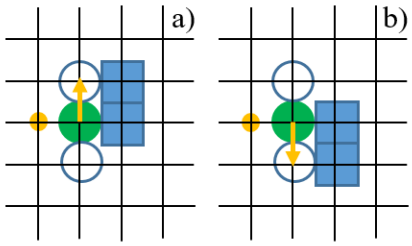


Figure 6. Moving-up (a) and moving-down (b) diffusion events.

The figure 7 shows some local configurations leading to forbidden diffusion (Fig. 7a) or evaporation (Fig. 7b, 7c and 7d) events. In each case, the symbol corresponding to the particle avoiding the movement of a mobile particle (the green particle) is bordered with red. In the fig. 7a all diffusion events are forbidden because the considered mobile particle has a top nearest neighbour. Of course, in such situation, even if not explicitly represented in the figure, the evaporation is forbidden as well. In all other situations described in the fig. 7b, 7c and 7d, the evaporation is also forbidden: because of a defect as a below nearest neighbour (Fig. 7b), because of the fact that a site on the trajectory (Fig. 7c) or a nearest neighbour of such site (Fig. 7d) is occupied by another particle. However, in these last three cases the mobile particle can still diffuse. According to the diffusion model described above, each possible movement for the mobile particle in fig. 7b is equivalent to a detrapment event. The diffusion events corresponding to the mobile particles in fig. 7c and 7d can belong to both in-layer or layer-to-layer diffusion event categories.

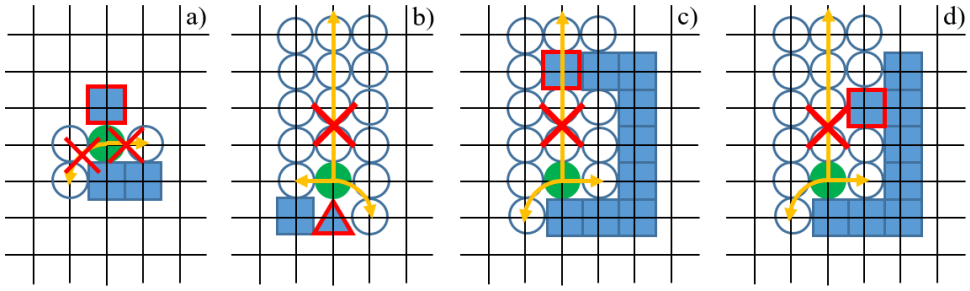


Figure 7. Possible local configurations leading to forbidden diffusion/evaporation events for some mobile particles (see in the text).

Finally, a 3D view of the main events implemented in NASCAM current version is shown in figure 8. As it can be seen, excepting the evaporation events, the activation energy corresponding to a given event is not depending on the nature of its nearest neighbours (the nature of surface below).

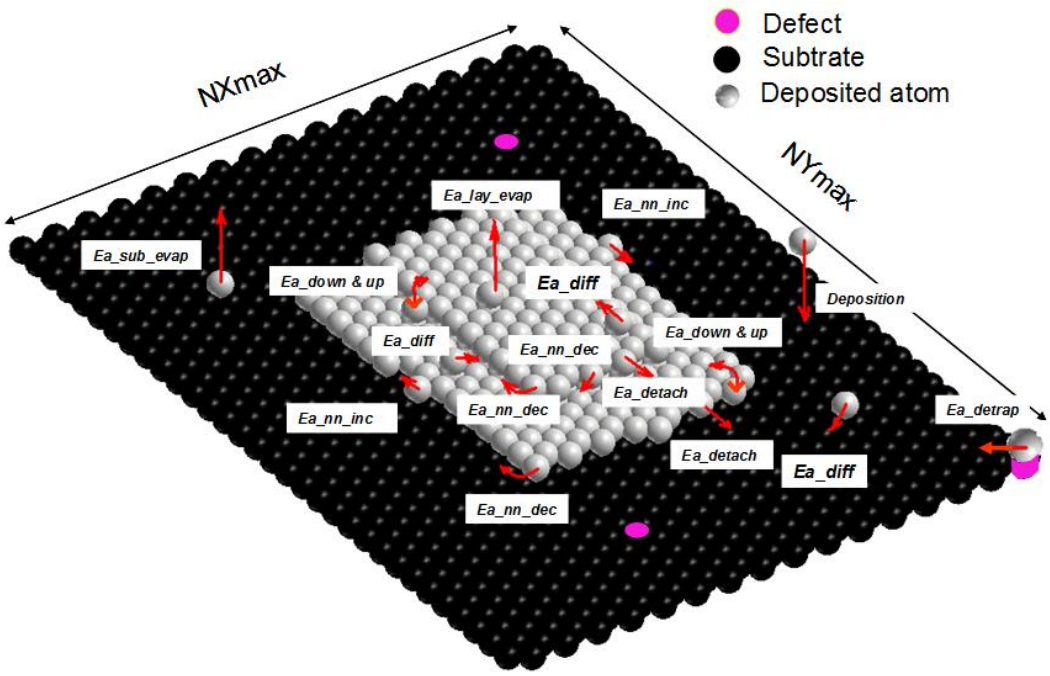


Figure 8. A 3D view of the main events implemented in the current version of NASCAM.

In what concerning the substrate, this can have either a hexagonal structure (suitable for amorphous or (111) surface) or a cubic structure. The substrate structure is fixed and does not evolve with time. So, NASCAM is not suitable to predict phase formation or transformation. It is also possible to start a simulation with a pre-existing surface structure such as a film or island present at the surface before deposition.

In the current version it is possible to have two types of atoms in the incident flow: a main specie supplying the system with mobile particles and a non-reactive gas specie bombarding the film during the deposition (e.g. Ar). In this case:

- Both species can have their own incident energy and angular distribution,
- The main specie particle attaches always to the surface of the already deposited structure, but the gas atoms may attach to the film only under special conditions (see below),
- Both kind of atoms can transfer energy to the film and to the substrate even if they do not attach to the system.

For the interaction of gas atoms with the film, several assumptions are made:

- They can be attached to the film only if they are trapped in a gas trap – an empty lattice site having all its lateral nearest neighbour sites occupied by other particles (substrate, mobile or gas particles),
- They can't move,
- They can be released from the film if the gas trap is opened because of evaporation or sputtering events (open trap – gas-trap as defined above having no particle as a top nearest neighbour and at least a lateral nearest neighbour position occupied by a gas or empty).

4.3 ACTIVATION ENERGIES CALCULATION

The activation energies for the events listed in figure 8 can be either found in literature or calculated by molecular dynamics or potential models. It is up to the user to estimate them. Fortunately, the NASCAM community will provide some of them. We describe here how we evaluate them for single elements.

Two models can be used for the calculation of activation energies. The first potential used is derived from the second moment approximation of the tight binding model (TB-SMA). The second potential is a 12-6 Lennard-Jones potential. The nudged elastic band (NEB) method is used in order to find the minimum energy path (MEP) followed by the atoms during surface diffusion (assumption).

4.3.1 NEB Method

The first step consists in building a substrate on which the considered atom is placed and the substrate exhibits terraces and steps for specific calculations.

At the second step, the initial and final positions are defined with respect to the activation energy that has to be calculated. *x* and *y* positions on the surface are defined by the crystallographic structure of the material and the *z* position is set to minimize the energy of the system at the final and initial positions.

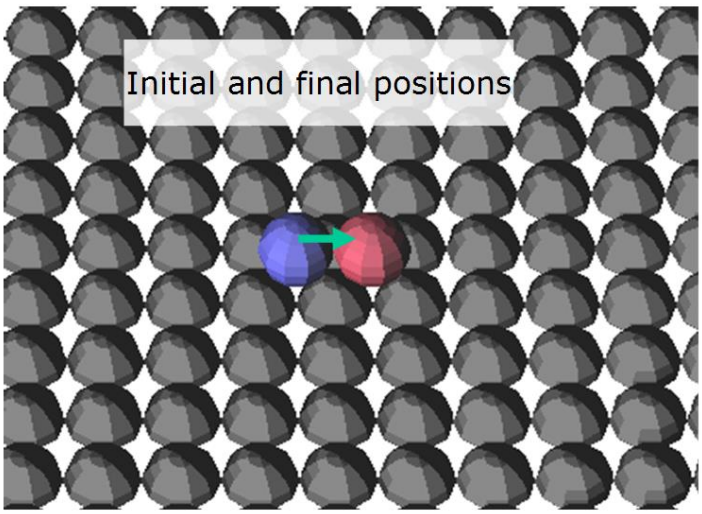


Figure 9. Initial and final positions used when considering free particle diffusion.

The images needed for the NEB method are then uniformly dispersed between the initial and final positions and the energy of the whole system is minimized.

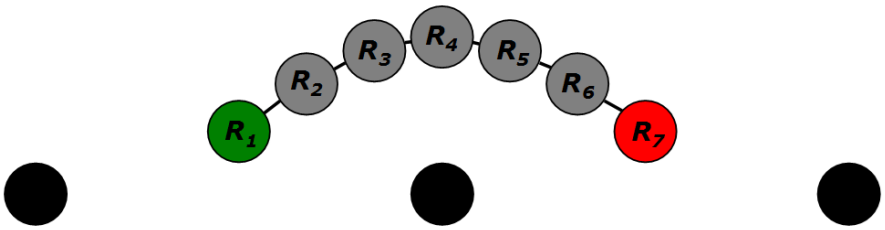


Figure 10. Images used in the NEB method for free particle diffusion (black – substrate, green and red - initial and final positions, grey - images).

The potential is then calculated along the MEP. The energy barrier can thus be estimated. The following figure shows an example for Cu/Cu(100).

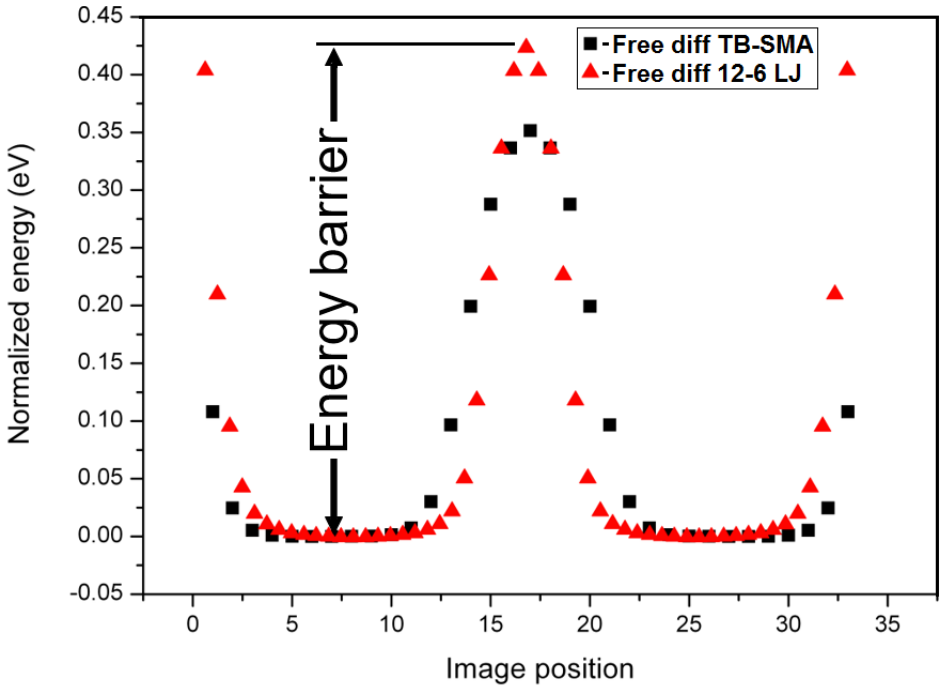


Figure 11. Potential energy curves along the MEP corresponding to the Cu free particle diffusion on Cu(100) using TB-SMA (black squares) and 12-6 L-J (red triangles) potentials.

4.3.2 Example: Cu/Cu(100) and Pt/Pt(100)

Two examples of a complete set of activation energies calculated using NEB method are given in Table 3 and Table 4.

Table 3. Cu/Cu(100)

Event	Name in NASCAM	TB-SMA (eV)	L-J 12-6 (eV)	Literature (eV)
Free diffusion	Ea_diff	0.351	0.424	0.36±0.03[1], 0.40[2], 0.48[3]
Diffusion along an island edge with inc. or equal nn number	Ea_nn_inc	0.239	0.404	Not found
Diffusion along an island edge with dec. nn number	Ea_nn_dec	0.253	0.406	Not found
Detachment from an island	Ea_detach	0.516	0.433	Not found
Jump up	Ea_up	0.716	0.499	Not found
Jump down	Ea_down	0.716	0.473	Not found

[1] Gert Ehrlich, Grazyna Antczak, Surface Diffusion, Cambridge University Press, 2010 pp 295-300
[2] J. J. De Miguel, A. Sánchez, A. Cebollada, J. M. Gallego, J. Ferrón, and S. Ferrer, Surface Science 189–190 (0), 1062 (1987).
[3] J. J. De Miguel, A. Cebollada, J. M. Gallego, J. Ferrón, and S. Ferrer, Journal of Crystal Growth 88 (4), 442 (1988).

Table 4. Pt/Pt(100)

Event	Name in NASCAM	TB-SMA (eV)	L-J 12-6 (eV)	Literature (ev)
Free diffusion	Ea_diff	0.835	0.7	0.44,1.25,0.47 [1]
Diffusion along an island edge with inc. or equal nn number	Ea_nn_inc	0.932	0.666	Not found
Diffusion along an island edge with dec. nn number	Ea_nn_dec	0.932	0.666	Not found
Detachment from an island	Ea_detach	0.888	0.719	Not found
Jump up	Ea_up	2.33	0.836	Not found
Jump down	Ea_down	2.33	0.79	Not found

[1] Gert Ehrlich, Grazyna Antczak, Surface Diffusion, Cambridge University Press, 2010 pp 295-300

Innovative Coating Solutions,
Place Saint-Pierre, 11, B-5380 Forville., Belgium
slu@incosol4u.com www.incosol4u.com

4.4 REACTIVE SPUTTER DEPOSITION

If one wants to simulate film growth by reactive sputter deposition (RSD) then he should introduce fluxes of reactive specie. The details of how one has to define the reactive flux to the substrate are given in the **Section 5.1 INPUT FILES**. Here a short description is given.

In most cases, RSD is performed in atmosphere of reactive gas, Oxygen for example. In this case the flux of oxygen may be calculated based on a kinetic theory of gases,

$$f = \frac{1}{4} n v_T = \frac{p}{\sqrt{2\pi m k T}}$$

where n and p are a gas concentration and pressure, T is its temperature, k is Boltzmann constant, and m is a mass of a molecule. In most cases this flux is much higher than a metallic flux, and it is possible to set a value of a reactive flux so that it is much higher than metallic flux. For example, if one set the metallic flux to 1, then reactive flux may be set to 100.

Beside the value of the flux one also has to set its energy and angular distributions. As reactive gas is thermalized in most cases, the one can set its mean energy equal to kT, or about 0.03 eV. As for angular distribution, one can suppose that reactive specie has uniform angular distribution

Finally, it is necessary to setup Stoichiometry index. Stoichiometry index (SI) is the maximum relative amount of the reactive specie in the compound. For example: SI = 1 for ZnO, SI = 1.5 for Al_2O_3, SI = 2 for TiO_2.

4.5 EVAPORATION

Re-evaporation of the deposited atoms may play an important role in the process of atomic deposition and in the further dynamics of the deposited film. For this reason, the number of physical processes the code deals with was extended to include the evaporation of the adatoms.

The simple analytical model, based on rate equation that takes into account attachment of adatoms to the existing islands and evaporation, may be written as follows:

$$dn/dt = -Q_{evap} - n\sigma ND + q_{dep} \qquad (9)$$

where n is the surface concentration of adatoms, q_{dep} is the deposition rate, Q_{evap} is the evaporation rate, N is a surface concentration of islands, and σ is a capture number which characterizes the diffusional flow of adatoms to islands, σ~*5-10*.

In a steady state regime

$$Q_{evap} + n\sigma ND = q_{dep} \qquad (10)$$

Innovative Coating Solutions,
Place Saint-Pierre, 11, B-5380 Forville., Belgium
slu@incosol4u.com www.incosol4u.com

as

$$Q_{evap} \sim n/\tau_{evap} = n\, w_o exp(-E_{evap} / k_B T), \quad (11)$$

and

$$D \sim d^2\, w_o exp(-E_{diff} / k_B T), \quad (12)$$

then one can derive the expression for the sticking coefficient β:

$$\beta = (q_{dep} - Q_{evap})/q_{dep} = (1 + 1/n\sigma N D\, \tau_{evap})^{-1} = \frac{1}{1 + a \cdot \exp(\frac{E_{diff} - E_{evap}}{k_b T})}, \quad (13)$$

where $a = 1/\sigma N d^2$, d is the inter-atomic distance.

One can see that the sticking coefficient depends not only on temperature and activation energy for evaporation, but also on diffusion. This can be studied with the help of NASCAM.

4.6 INCIDENT ENERGY DISTRIBUTION

The user can specify the energy of the atoms being deposited. Depending on the nature of the collided particle - the target, an energetic incoming particle - the projectile, can produce different local modifications of the system structure:

1. if the target is a substrate particle and if the energy of the projectile exceeds the energy needed to produce a defect, then a reversible local "surface trap" is generated. This surface trap is a local defect without being an empty position – the substrate particle keeps its lattice position. Any mobile particle placed in the layer above such surface trap and having it as a nearest neighbour or as a next nearest neighbour will be "trapped". The probability that adatoms break the bond and leave the trap, diffusing on the terrace or going down a step, depends on the de-trapping associated energy barrier Ea_detrap.

2. if the target is a gas or mobile particle and if the amount of energy transferred from the projectile to the target is larger than a well-established energy threshold, then the target together its nearest and next nearest neighbours can, if some other supplementary conditions are fulfilled as well, become recoil particles and move to other lattice sites or can be removed from the system. In both cases the projectile spends all its kinetic energy and sticks to the system surface according to the rules imposed by its own nature, gas or mobile particle (see above).

NOTE. *The code can run incorrectly if y-dimension of the substrate is less or equal to 2.*

In the code the energy transfer is calculated according to binary elastic collision approximation. This approach is well known and one can find a brief description here (see for example [6])**:**

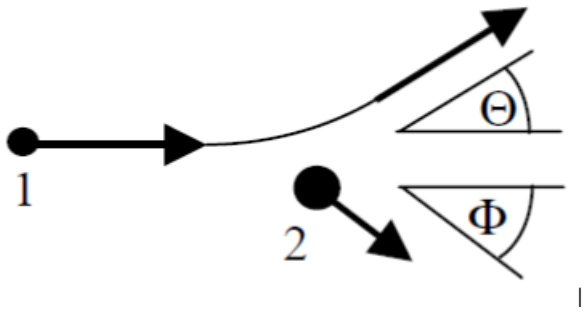


Figure 12. A geometrical description of a collision process: 1 - incoming particle (projectile), 2 - particle belonging to the system (target), Φ - recoil angle and Θ - scattering angle.

The transferred energy is given by:

$$\Delta E = E\frac{4m_1m_2}{(m_1+m_2)^2}\sin^2\left(\frac{\theta}{2}\right), \qquad (14)$$

where m_1 and m_2 are masses of projectile and recoil particles and θ is the scattering angle in the centre-of-mass system:

$$\theta = \pi - 2b\int_{o}^{R_m} \frac{dR}{R^2\sqrt{1-\frac{U(R)}{E_c}-\frac{b^2}{R^2}}}, \qquad (15)$$

where b is an impact parameter, R_m is the minimum distance between the atoms, and E_c is the energy in the centre-of-mass system:

$$E_c = E\frac{m_2}{m_1+m_2} \qquad (16)$$

The scattering angle in the laboratory system can be calculated as:

$$\Phi = \frac{\pi-\theta}{2}. \qquad (17)$$

Collision cascade is not taken into account. If the recoil particle energy is larger than its bounding energy corresponding to its local configuration it can move to the lattice site closest to its trajectory as long as this site is an empty site. If this site is already occupied by another particle the recoil particle does not leave its initial site. For each recoil particle this "closest lattice site" is one of its nearest and next nearest neighbour sites. If on its new position the recoil particle becomes an isolated particle then it is removed from the system.

The incident energy E is taken from the energy distribution of the incident atoms and the impact parameter b is calculated taking into account the trajectory of the projectile and the initial position of the target.

For the energy distribution of the incident atoms, either a constant value (E_0), an analytical build-in function, or external data can be used. Currently, the build in energy distribution is:

$$\rho(E) = \frac{\exp(-E/E_0)}{E_0}, \qquad (18)$$

where E_0 is the average energy of the incident atoms (in eV). Figure 13 shows an example with E_0 = 5 eV.

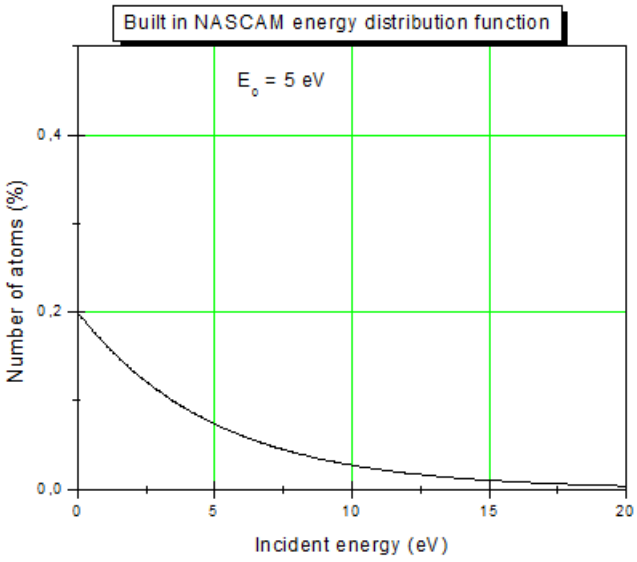


Figure 13. Example of the implemented energy distribution function probability with mean energy = 5 eV.

Other energy distribution data can also be loaded into NASCAM, giving the possibility to use either experimental data or data calculated by software like SIMTRA [7]. Details on how to load data are given in the section "Energy distribution data file".

The energy of an incident atom is selected stochastically with accordance to its energy distribution $\rho(E)$. If w is a random number from the interval [0; 1] then the energy of an incident atom E can be found from the following equation:

$$w = \frac{1}{A}\int_0^E \rho(x)dx\,, \qquad (19)$$

where

$$A = \int_0^\infty \rho(x)dx\,. \qquad (20)$$

We choose a random number w, and then we solve equation 19 for that random number. The root of the equation is the energy of the incident atom.

4.7 INCIDENT ANGULAR DISTRIBUTION

It is possible to set the angular distribution of incident atoms by three different means: a data file, a build in distribution function, or as a uniform angular distribution. The last may be used to describe an angular distribution of a chamber gas.

If set by a distribution function the incident angles are specified by a function and two angles. The geometry of the deposition is shown in the following figures. The mean incident direction is defined by the pair θ_0 and φ_0 (fig. 14a, blue arrow). The angular distribution of the incident atoms around the mean incident direction is given by the following functions:

$$f(\theta) = \exp(-\xi^2 / 2\Delta\theta^2), \qquad (21)$$

where Δθ is a mean deviation from the mean direction (fig. 14b). We suppose that the distribution of the incident atoms around the mean incident direction is uniform in probability, i.e. φ and ξ are chosen randomly with the same probability.

For fixed angle deposition, one should specify θ_0 and φ_0 and fix $\Delta\theta = 0$.

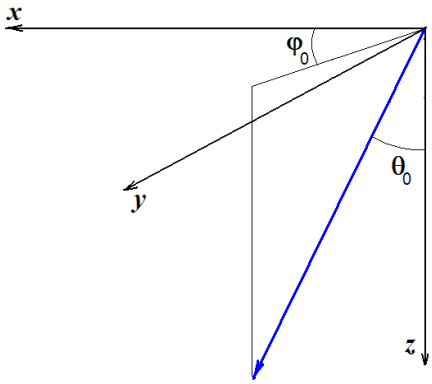


Figure 14a. Angular coordinates θ_0 and φ_0 of the mean direction (blue arrow) of the incident particles.

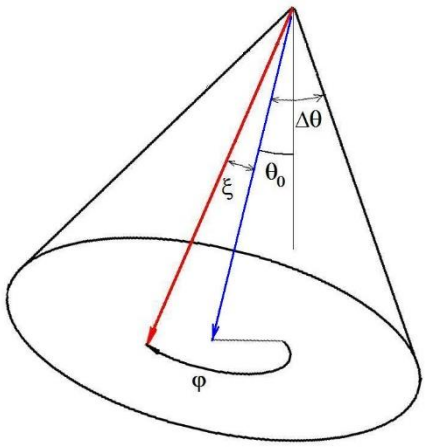


Figure 14b. The scheme of distribution of particles (red arrow = 1 particle) in a cone.

In figures 15a and 15b one can see two distributions of the sites on the substrate where incident atoms landed. 200 atoms were emitted from the point (0, 0, 1) in the direction $\theta_0 = 30°$ and $\varphi_0 = 0°$ with $\Delta\theta = 5°$ (Fig. 15a) and $\theta_0 = 45°$ and $\varphi_0 = 20°$ with $\Delta\theta = 5°$ (Fig. 15b). As the particles are uniformly distributed in a cone, they draw an ellipse on the surface.

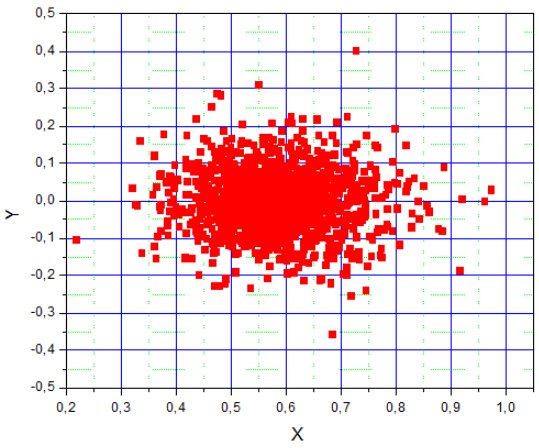


Figure 15a. Top view of the landing sites of incident particles, $\theta_0 = 30°$ and $\varphi_0 = 0°$, $\Delta\theta = 5°$.

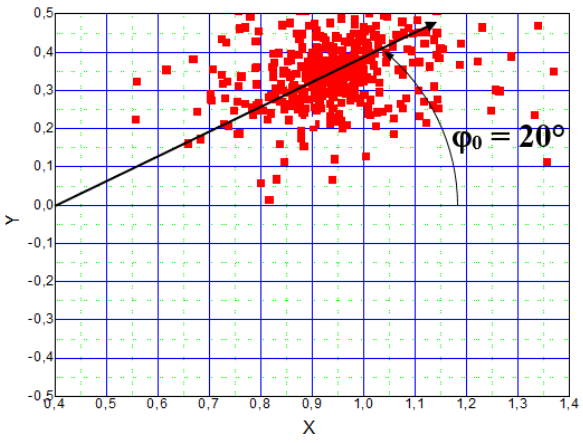


Figure 15b. Top view of the landing sites of incident particles, $\theta_0 = 45°1$ and $\varphi_0 = 20°$, $\Delta\theta = 5°$.

Figures 16a and 16b show an example of particle distribution in the cone for $\theta_0 = 30°$, $\varphi_0 = 0°$ and $\Delta\theta = 5°$.

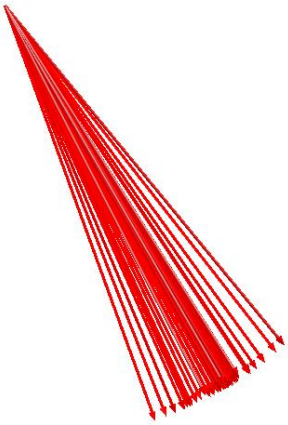


Figure 16a. The distribution of particles in the cone, view from y direction.

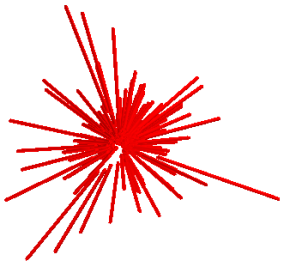


Figure 16b. The distribution of particles in the cone, view from mean incident direction.

During the simulation, the origin of the cone, i.e. the starting point of atoms being deposited, is chosen randomly over the defined sample surface.

The other option is that NASCAM reads the angular distribution from a file. This may be used in conjunction with SIMTRA.

4.8 MASKED DEPOSITION

It is possible to define a mask through which the particles will travel and will be deposited on the substrate. The mask is defined by a set of rectangular openings whose geometric definitions are included in a text file, as four integer numbers x_0, y_0, *dx* and *dy*. The name of this file is given as an input value for Pattern_name field in the *input.txt* file. Each line in the mask definition file contains four numbers giving a complete definition of a slit in a (x, y) plane: first two numbers, x_0 and y_0, correspond to its left bottom corner and the third and fourth number, *dx* and *dy*, correspond to its dimensions (in lattice parameter units) on x and y directions. Obviously, the mask definition parameters must be chosen in such a way that the entire mask size is smaller than the size of the substrate. It is possible to have several masks and they may overlap. The following example defines an L like mask, see figure 17.

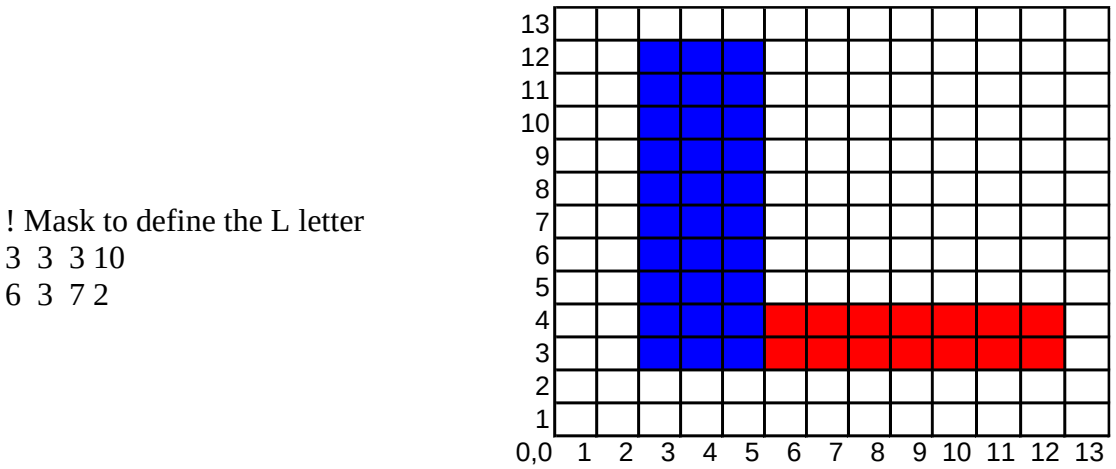


Figure 17. A mask defining a capital L: on the feft - the content of the mask definition file and, on the right - the corresponding mask representation in an (x, y) plane.

To switch masked deposition on it is necessary to set the Masked_deposition flag to 1 in the input.txt file:

Masked_deposition	1
Pattern_name	pattern.txt

One should note that the in the z-direction a mask is positioned right above the growing film. That means that no atoms can be deposited outside the mask boundary.

4.9 SOURCE LINEAR MOVEMENT

As it was mentioned above, the atom source can be collimated but also animated with a linear movement. The deposited atoms can reach the surface passing through a mask (collimator) which moves at the same velocity as the source (see Fig.18).

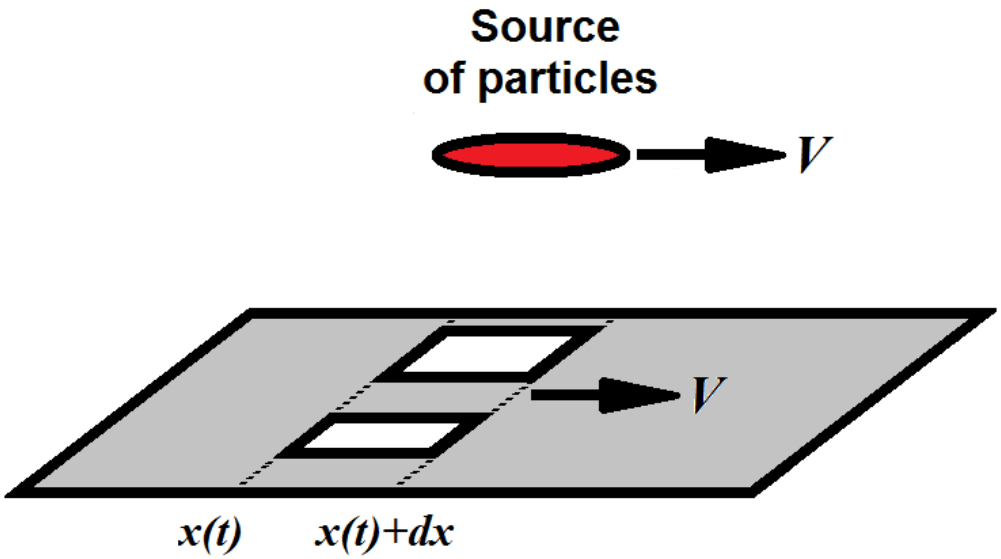


Figure 18. Deposition onto a moving substrate (or on a fixed substrate with a moving source): note that there are two sub-masks, both having the same initial position x_0 and

length *dx*. The speed of masks being *V*, the *x(t)* position of masks/source depends of time *t* as $x(t) = x_0(t) + V \times t$.

To simulate deposition on a moving substrate, it is necessary to

- set both flags Masked_deposition and Source_linear_movement to 1,
- define a mask in a file (e.g. pattern.txt),
- set the velocity (second number for Source_linear_movement field - in lattice parameter units per second):

Masked_deposition	1
Pattern_name	pattern.txt
Source_linear_movement	1 0.05

Several sub-masks can be used but they must have the same x_0 starting coordinate and the same length *dx* (see Fig. 18).

It is compulsory to specify carefully the deposition rate (*F*), number of deposited mobile particles (*Ndep*), x-size of the substrate (*NXmax*), the initial position, dimensions and the velocity of the mask movement (x_0, *dx* and *V*). Atoms are deposited onto the substrate as long as the entire surface of mask is within the substrate surface:

$$x_0 + dx + V \times t < NXmax - 1,$$

where *t* is the current deposition time determined by the deposition rate, substrate dimensions and number of already deposited atoms. If the above condition becomes false the simulation stops even if not all requested particles (*Ndep*) have been deposited.

4.10 SOURCE TILT AND ROTATION

It is also possible to simulate deposition with a rotating source (or on a rotating substrate), The vertical axis can also be subject to an oscillation movement (see figures 19 to 24). This has been implemented to simulate Glancing Angle Deposition (GLAD, see the Examples manual)

There are four parameters to specify:

w_rot(1/s);init_tilt(deg);A_osc(deg);w_osc(1/s) 0.0 0.0 0.0 0.0

The first one is the angular velocity of the rotation along the axis source (w_rot, 1/s), the second is the initial tilting of the substrate (Ω_0), the third is the amplitude of the oscillation (A, deg) and the fourth is the oscillation frequency (w_osc, 1/s).

Amplitude and oscillations are related through:

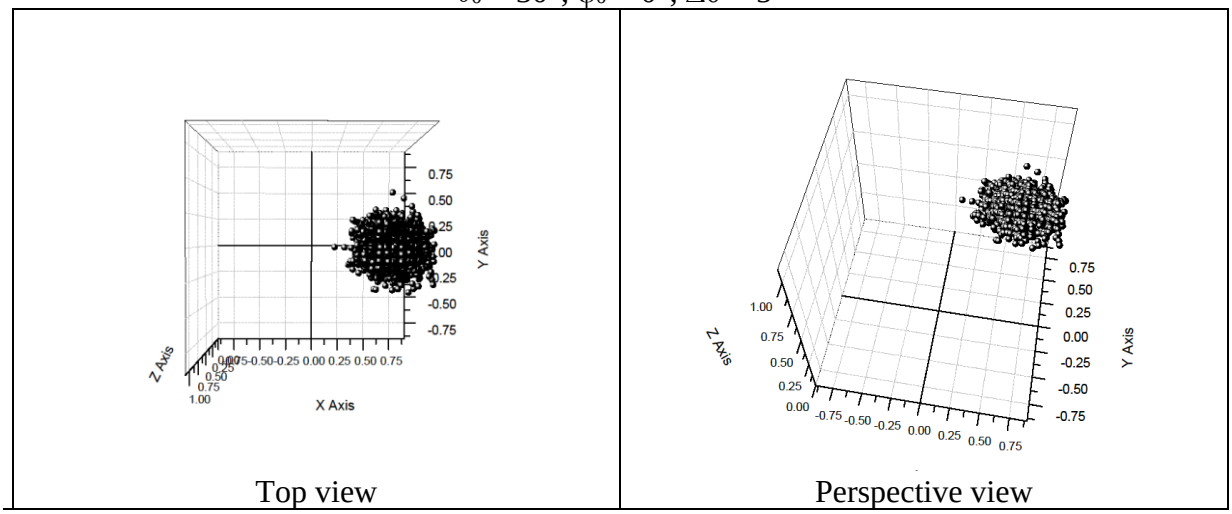
$$\Omega = \Omega_0 + A*sin(w\_osc*t)$$

4.10.1 EXAMPLES OF SOURCE MOVEMENTS:

The figures 19 to 24 give some examples of oscillating sources. For all cases the origin of the deposited atom is shown as if it would be seen from the (0,0) location on the substrate (like an avatar located in 0,0 and looking to the sky). In the following examples, for "Broad angle", "Rotation" and "Tilting" cases, the broadening of the beam is $\Delta\theta = 5^{o}$. For "Tilting plus Rotation" cases, the broadening of the beam is $\Delta\theta = 0^{o}$ for better visualization of the "starting" region

4.10.1.1 Broad angle

$\theta_0 = 30°$, $\varphi_0 = 0°$, $\Delta\theta = 5°$



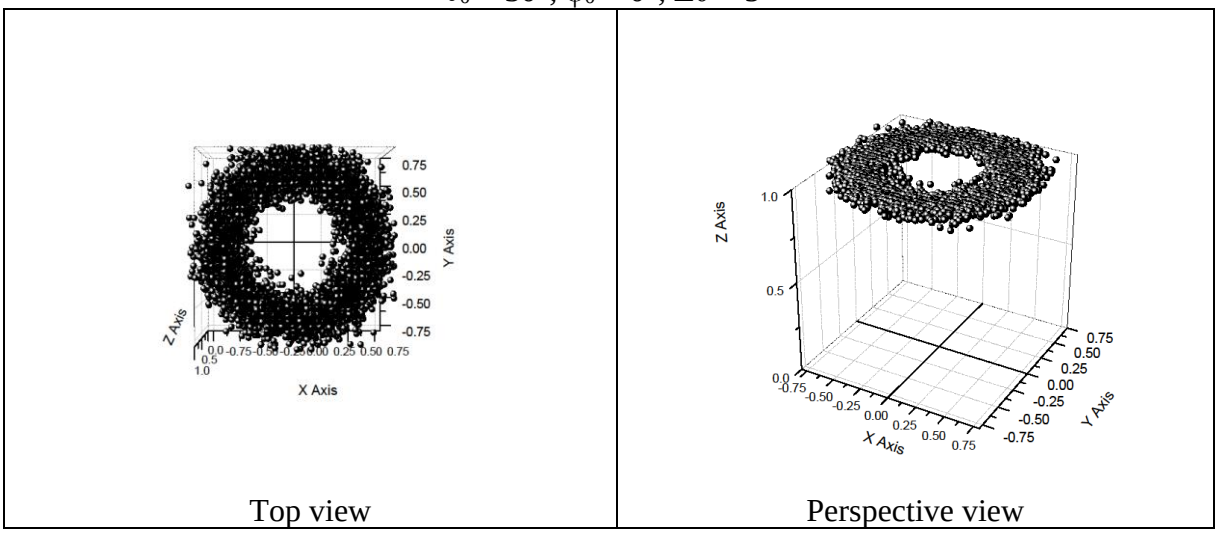
Top view | Perspective view

Figure 19.

w_rot(1/s);init_tilt(deg);A_osc(deg);**w_osc**(1/s) **0.0** 0.0 0.0 **4.5**

4.10.1.2 Rotation

$\theta_0 = 30°$, $\varphi_0 = 0°$, $\Delta\theta = 5°$



Top view | Perspective view

Figure 20.

w_rot(1/s);init_tilt(deg);A_osc(deg);**w_osc**(1/s) **0.57** 0.0 0.0 **4.5**

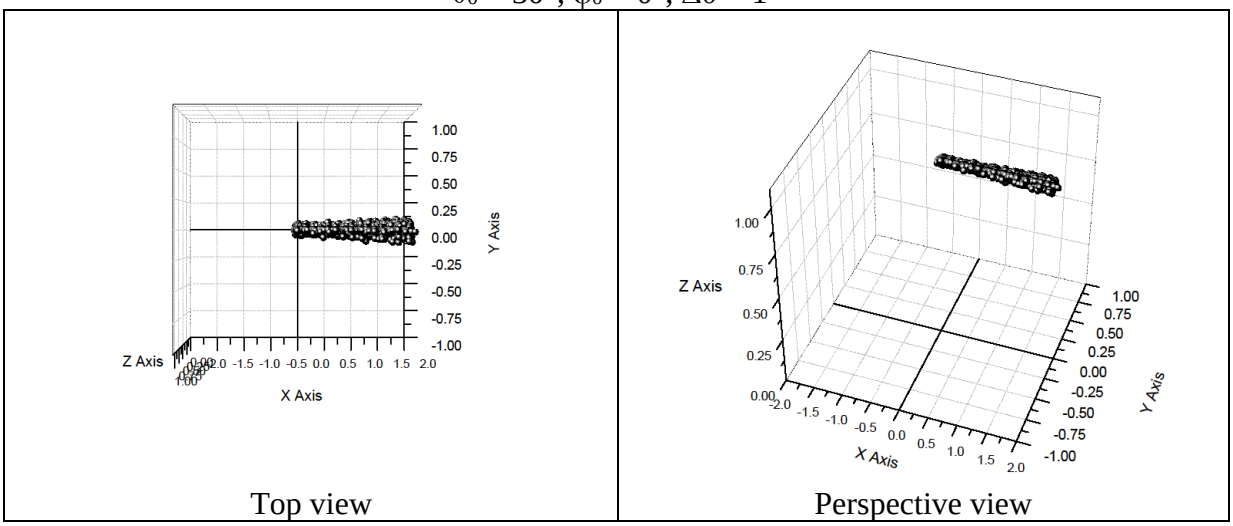
Innovative Coating Solutions,
Place Saint-Pierre, 11, B-5380 Forville., Belgium
slu@incosol4u.com www.incosol4u.com

4.10.1.3 Tilting.

Note: Tilting is always around Y axis

Beam along X

$\theta_0 = 30°$, $\varphi_0 = 0°$, $\Delta\theta = 1°$



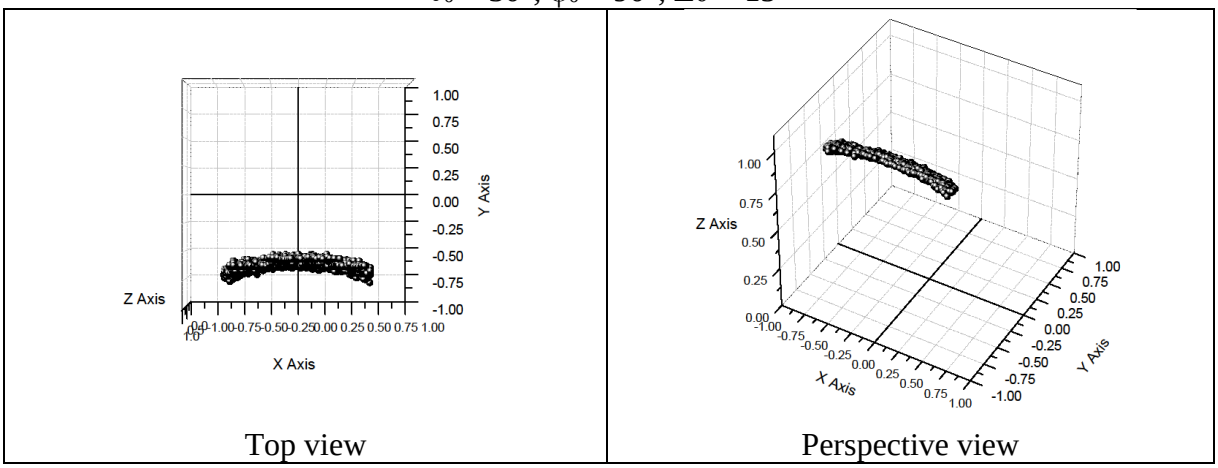
Top view | Perspective view

Figure 21.

w_rot(1/s);init_tilt(deg);A_osc(deg);w_osc(1/s) 0.0 0.0 30 4.5

Beam along Y

$\theta_0 = 30°$, $\varphi_0 = 90°$, $\Delta\theta = 15°$



Top view | Perspective view

Figure 22.

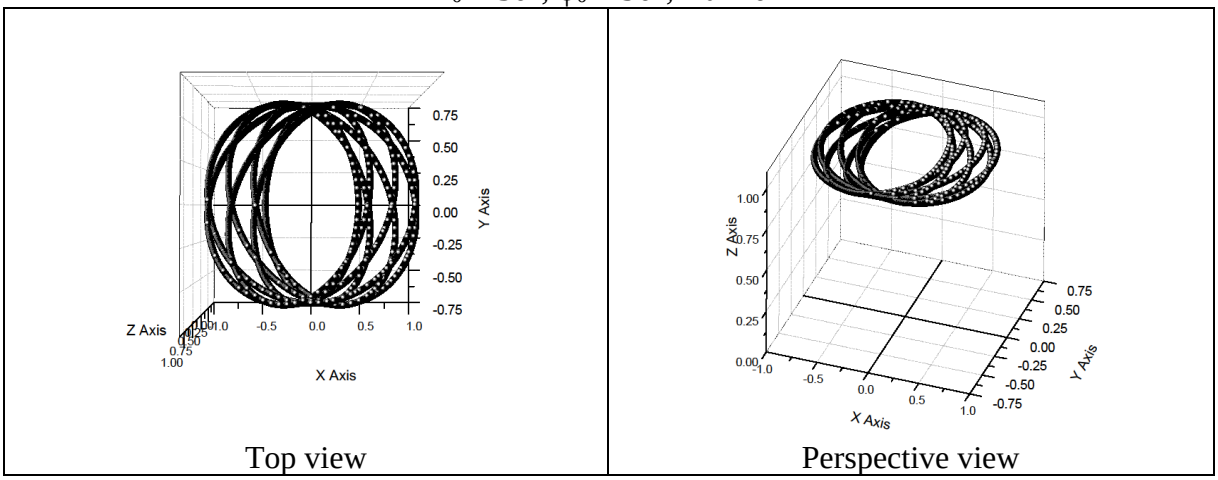
w_rot(1/s);init_tilt(deg);A_osc(deg);w_osc(1/s) 0.0 0.0 30 4.5

4.10.1.4 Source tilt plus rotation.

If the broadening of the beam is large enough, there will be no difference between fast rotation and fast tilting.

Fast rotation

$\theta_0 = 30°$, $\varphi_0 = 90°$, $\Delta\theta = 0°$



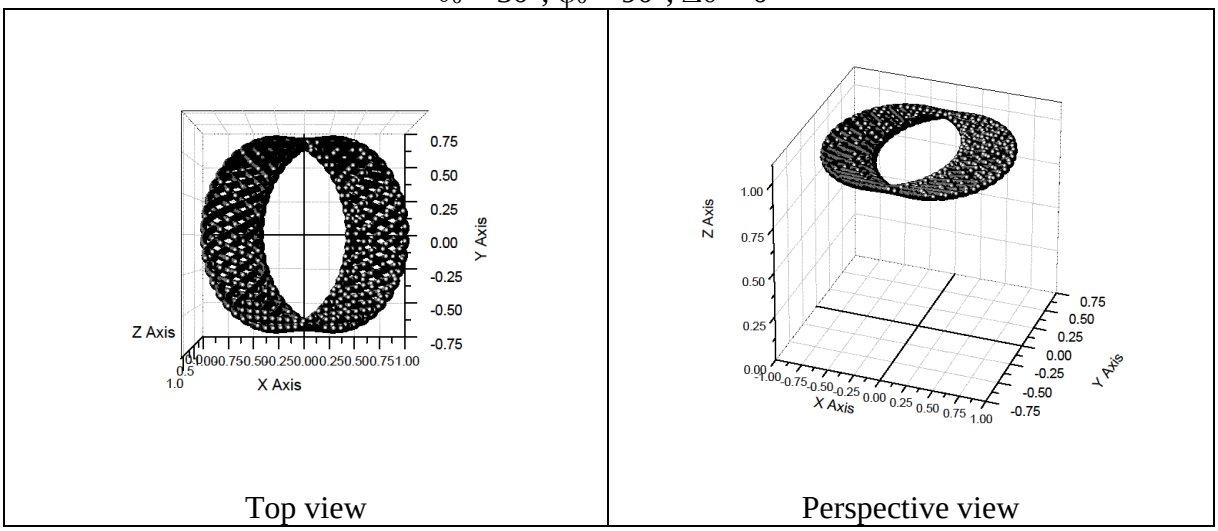
Top view | Perspective view

Figure 23.

w_rot(1/s);init_tilt(deg);A_osc(deg);w_osc(1/s) 0.7 0.0 10 0.5

Fast tilting

$\theta_0 = 30°$, $\varphi_0 = 90°$, $\Delta\theta = 0°$



Top view | Perspective view

Figure 24.

w_rot(1/s);init_tilt(deg);A_osc(deg);w_osc(1/s) 0.7 0.0 10 4.5

4.11 CODE STRUCTURE

The code consists of several logical blocks:

1. Initialization
2. Main part
3. Final part

4.11.1 Initialization

Once the user runs the code, the following parameters are initialized:

1. Size of the simulation box NXmax*NYmax*NZmax (NXmax, NYmax and NZmax are specified in lattice parameter units in the input.txt file)
2. Number of mobile particle to be deposited and annealing time (specified in the input.txt file)
3. Activation energies for all possible events and simulation temperature (specified in the input.txt file).
4. Particular deposition parameters for each deposited specie (mobile and gas particle): relative flow values, specie names and atomic masses, information regarding their energy and angular distributions (specified in the iput.txt file or, if requested for energy and angular distributions, read from a specific file).
5. Surface binding energies for substrate and film (specified in the input.txt file)
6. Parameter for forced deposition (specified in the input.txt file)
7. Configuration of the substrate (optional - defined in the substrate.xyz file)
8. Initial positions of adatoms (optional - read always from the substrate.xyz file)

The NZmax dimension of the simulation box is increased automatically if the coating thickness increases above the starting value.

4.11.2 Main part – kMC loop

In this part, the system evolves according to the model described above. Each kMC step consists in several sub-steps:

1. The mobile particles are sorted and placed in the corresponding event lists according to the type of the movement in which they can participate. These event lists are created in the beginning of each new simulation and only updated at each kMC step. As the type of the event is defined by a local configuration around the particle it is not necessary to update these lists for all the particles as for most of them their neighbouring remains the same after the pervious time step. So the event lists are updated only for the particles which neighbouring has been changed. The updated region is a surrounding of a particle moved at a previous step or of a newly deposited particle.
2. Choose an event for the current simulation step according to its computed probability.
3. Choose an atom from the list corresponding to the previous selected event and move this atom to its new position (lattice site).
4. Calculate a time increment and increase accordingly the simulation time and deposition timer.

5. Perform the deposition of a new atom at the appropriate deposition timer value.

4.11.3 Final part

At the end of a simulation several structure files are created within the current folder (or output folder when GUI is used – see GUI manual too):

- *coating.xyz* - this file contains the coordinates of substrate and all deposited atoms in the system as well as their names (attention: Xx notation corresponds to a defect – "surface trap").
- *deposited_layer.xyz* - this file contains the coordinates of the adatoms deposited in the last deposition session (does not contain substrate atom coordinates).
- Intermediate structure snapshots are saved in the *film_growth.xyz* file upon the user request.

The atom coordinates in all structure files (having the extension .xyz) are given in lattice parameter units. To have the real values of these coordinates it is necessary to multiply their output values by the lattice parameter value in real length units (Å or nm). To visualize/analyse the output structure files (*substrate.xyz, film_growth.xyz, coating.xyz* & *deposited_layer.xyz*) JMOL package can be used.

A certain number of .csv files containing the information about deposition, film growth etc (see sections 6.2.4 – 6.2.8) are created in the beginning and then are updated in the course of the simulation.

5 Usage of the code

5.1 INPUT FILES

To use the code it is compulsory to have in the current directory, together the executable Nascam.exe, at least the *input.txt* file. Other files are optional (e.g. *substrate.xyz* or energy/angular distribution files) providing they are not used. Input files may include a "comments", placed before all data lines. Each comment line must start with '!'. One can run Nascam.exe with the switch "*–n filename*" in order to use file "*filename*" instead of "*input.txt*"

5.1.1 input.txt

This is the most important input file. It is a.txt file that can be edited by any text editor (Fig. 25). Because NASCAM reads only the numbers at the end of each line, the text (line description) at the beginning of each line can be modified according to user needs. However, the order of the parameters (lines) has to be preserved as listed and the line descriptors should not contain any spaces.

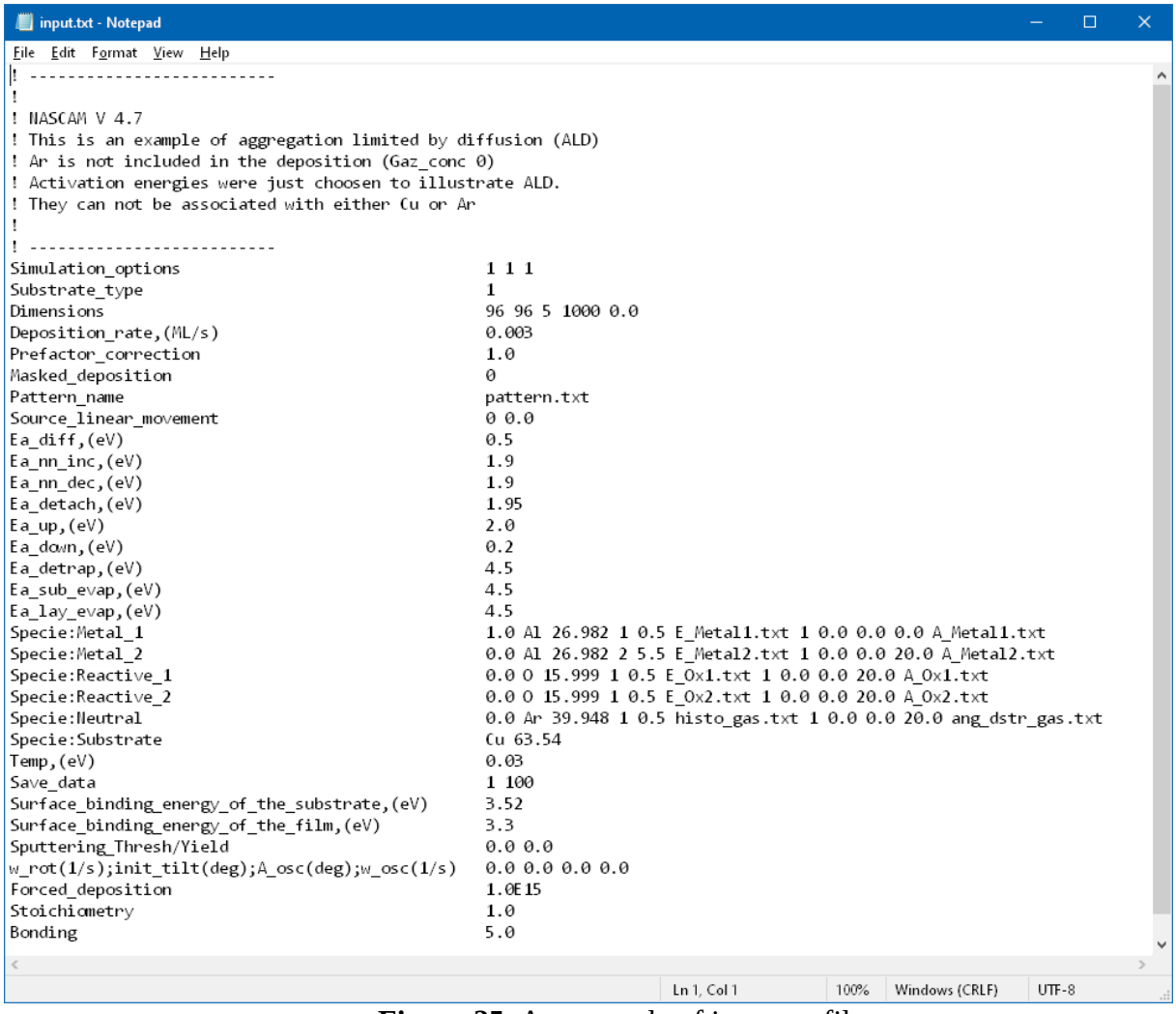


Figure 25. An example of input.txt file.

A detailed description of each control parameter specified in input.txt file is given below:

Simulation_options:

The first parameter, *n*, specifies how often the information about the current status of the simulation is printed out to the screen and log file as well as how often the statistics regarding the film growth (film roughness, density, number of islands etc) are calculated. The value of *n* means that the data printed/calculated every *n* deposited atoms. This option may be especially helpful in case of deposition of large amount of atoms. Then the sizes of output files are reduced and the execution time is also reduced due to less output to the screen/disk.

The second parameter is a diffusion computation switch: 0 - no diffusion or evaporation (no update of the lists of events is done during the current simulation) 1 – take diffusion and evaporation into account.

The third parameter is also a switch controlling the island statistics calculation: 0 - no calculation, 1 – calculate number of islands and their size distribution. If this analysis is requested (value set to 1) then it is done with the frequency established by the first parameter value *n*.

Substrate_type:

Two values are possible for this structure switch: 1 for a hexagonal structure or 0 for a cubic structure (Fig. 26):

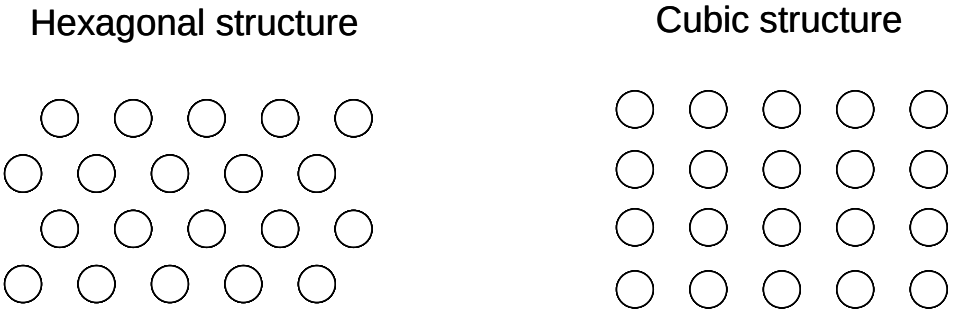


Figure 26. Sketch of the two possible substrate atomic structures.

Dimensions:

The first three numbers define the X & Y & Z system dimensions (*NXmax*, *NYmax* and *NZmax* in lattice parameter units). In addition to define the simulation box dimensions they are also used to allocate the memory dynamically.

Periodic boundary conditions are used on both X and Y directions. In the case of hexagonal symmetry (Substrate_type switch value is 1 – see before), to ensure the validity of periodic boundary condition on y direction, it is compulsory that *NYmax* has an even value. The maximum number of substrate atoms and deposited atoms just depends on the physical memory of your computer.

NZmax is the maximum possible number of layers (including substrate, layers from previous depositions, etc). This is adjusted automatically during the simulation if the deposited structure thickness reaches that limit.

Next, set the number of mobile particles to be deposited during the current simulation *Ndep*, and eventually the duration of a post-annealing process *Tmax* (in seconds). For example, if you need to simulate a deposition process without any additional annealing time, it is

necessary to set *Tmax* to 0. If you need to simulate the annealing of an existing atomic configuration during a certain period of time, then it is necessary to set *Ndep* to 0. If you want to perform deposition and subsequent annealing at the same temperature, then you should set *Ndep* > 0 and *Tmax* > 0. If, for example, *Ndep* = 100 and *Tmax* = 100, first 100 atoms will be deposited. Second, the system will be annealed for 100 seconds. This is convenient for simulating deposition followed by annealing if both happen at the same temperature[1].

Deposition_rate,(ML/s):

The fourth line of *input.txt* is for the deposition rate given in monolayer per second. Knowing this value it is possible to determine the characteristic time of the system τ (the time interval between two successive deposition events - see eq.7 above):

$$\tau = \frac{1}{F \cdot \dim X \cdot \dim Y},$$

According to this equation a very large deposition rate prevents atom to diffuse because τ becomes too small.

Prefactor_correction:

An event rate is given by $w = w_0 \cdot exp(-E/k_B T)$ and for a single atom the value of the pre-exponential factor is of the order of the lattice vibrational frequency $w_0 \sim 10^{13}$ s^{-1}. But for the movement of clusters of atoms this value might be different. NASCAM may take into account the cluster diffusion using a new parameter: *Prefactor_correction*. This parameter is equal to the ratio of the desired value for the pre-exponential factor and the single atom diffusion pre-exponential factor. So, if one wants to have $w_0 = 10^8$ s^{-1} then the *Prefactor_correction* is equal to 10^{-5}.

Prefactor_correction 1e-5

Masked_deposition & Pattern_name:

When the switch *Masked_deposition* is set to 1, NASCAM reads the pattern (mask) as defined in the ASCII file having the filename given as value for *Pattern_name* input variable.

Linear Movement:

This line defines an optional linear movement of the substrate during the deposition.

First parameter has to be set to 1 for linear movement. Second parameter specifies the linear speed (in lattice parameter units per second).

Source_linear_movement 1 1.8

Ea_diff,(eV);Ea_nn_dec,(eV) ….Ea_lay_evap ,(eV):

The next several lines are dedicated to the activation energies for different kind of movements. All the values are given in eV.

These input values are not relevant as long as the simulation is performed at a temperature which is low enough so that no thermally activated events should take place, T << Ea's.

[1] If both do not happen at the same temperature, the user can first simulate the deposition at deposition temperature, and then use the data in coating.xyz as input data (substrate) for annealing at a different temperature.

The effect of setting the temperature to a low value is almost the same as to set up the simulation without diffusion by setting the second parameter of the *Simulation_options* line in input.txt file to 0: *1* ***0*** *1*.

However, there is a difference in what concerning the computation time: setting the second parameter for *Simmulation_options* to 0 avoids any updates of the event lists while performing simulations at very low temperature keeping this parameter to 1 such updates are still done. Of course, they are not so often, because the main event that happens is the deposition event…

In the current version there are several kinds of possible events. A complete description of them has been already done (see above). Here we only mention them adding some more details considered to be necessary for a better understanding:

1. free mobile particle diffusion (*Ea_diff*) - in-layer diffusion event
2. the diffusing particle having in the final position the same or larger number of lateral nearest neighbours than in its initial position (*Ea_nn_inc*) - in-layer diffusion event (Fig. 26a)
3. the diffusing particle having in the final position a lateral nearest neighbours number lower than in its initial position (*Ea_nn_dec*) - in-layer diffusion event (Fig. 26a)
4. the diffusing particle detaches from a step edge becoming a free particle (*Ea_detach*) - in-layer diffusion event
5. up diffusion event (*Ea_up*) - layer-to-layer diffusion event (Fig. 26b)
6. down diffusion event (*Ea_down*) - layer-to-layer diffusion event (Fig. 26b)
7. detrapment event (*Ea_detrap*) - in-layer/layer-to-layer diffusion event
8. evaporation event, the evaporating particle having as a bellow nearest neighbour a substrate particle (*Ea_sub_evap*),
9. evaporation event, the evaporating particle having as a bellow nearest neighbour a gas or mobile particle (*Ea_lay_evap*).

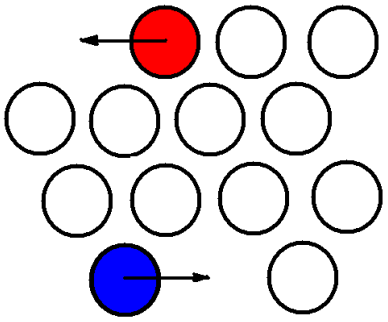


Figure 26a. Illustration of nearest neighbour increase (blue) and nearest neighbour decrease (red) movements associated to Ea_nn_inc and Ea_nn_dec**.**

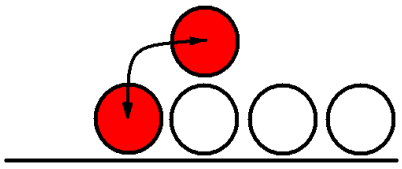


Figure 26b. Illustration of up and down movements associated to Ea_up and Ea_down**.**

Generally, the activation energies should satisfy the next relationships:

Ea_up > Ea_down > Ea_nn_inc, Ea_nn_dec, Ea_detach > Ea_diff,

and

Ea_detach ≥ Ea_nn_dec.

Species:

In the next three lines the details about deposited specie, chamber gas, and substrate are given.

Example:

Specie:Metal *1.0 Al 26.982 1 0.5 histo.txt 1 0.0 0.0 0.0 angDstr.txt*
Specie:Gas *0.0 Ar 39.948 1 0.5 histoGas.txt 1 0.0 0.0 0.0 angDstrGas.txt*
Specie:Substrate *Cu 63.54*

Specie:Metal_1 *1.0 Al 26.982 1 0.5 E_Metal1.txt 1 0.0 0.0 0.0 A_Metal1.txt*
Specie:Metal_2 *0.0 Al 26.982 2 5.5 E_Metal2.txt 1 0.0 0.0 20.0 A_Metal2.txt*
Specie:Reactive_1 *0.0 O 15.999 1 0.5 E_Ox1.txt 1 0.0 0.0 20.0 A_Ox1.txt*
Specie:Reactive_2 *0.0 O 15.999 1 0.5 E_Ox2.txt 1 0.0 0.0 20.0 A_Ox2.txt*
Specie:Neutral *0.0 Ar 39.948 1 0.5 histo_gas.txt 1 0.0 0.0 20.0 ang_dstr_gas.txt*
Specie:Substrate *Cu 63.54*

In the current version, there may be several species in the incident flow; two different species of depositing material, one reactive specie, and atoms of chamber gas. In the lines "*Specie:Metal_1*",… , "*Specie:Gas*" one should specify the relative amount of these species in the deposition flux, their names and atomic masses, details regarding their energy and angular distributions. Note, that although there are two lines, which describe the flux of a reactive specie, one must use the same specie for the both flows. Generally, the first line is used for the neutral atoms and the second line is used to describe ions. The same is valid for the metallic specie. The total flux may be divideded into two parts, one is neutral, the second is charged.

Syntax for both species is the same:

X_s $Name_s$ M_s Energy_source Mean_energy_of_incident_atoms Energy_file Angular_source Theta0 phi0 dtheta angular_file

where

X_s : relative amount of specie "s" in the flux, s = metal or gas. If X_{metal} = 2 and X_{gas} =1 then there is 66.6% of metal atoms and 33.3% of gas atoms in the incident flux. The fraction of each species is given by expression $X_s/(X_{metal} + X_{gas})$;

$Name_s$ M_s : the name and atomic mass of each specie;

Energy_source, *Mean_energy_of_incident_atoms*, *Energy_file*: The value of *Energy_source* is used to indicate how the information about the energy of incident particles is supplied to the program. Value of the switch is: **0** - for energy distribution read from a data file called *Energy_file* (see later for its detailed description), **1** - for constant energy given as *Mean_energy_of_incident_atoms* and **2** - for computed values using the analytical function $f = exp(- E/E_0) / E_0$. In the last case, the parameter *Mean_Energy_of_ incident_atoms* is for the mean energy of deposited atoms E_0. If the *Energy_source* switch is set to **1** or **2**, the data file *Energy_file* is ignored (and may be absent from the current folder).

Angular_source, Theta0, phi0, dtheta, Angular_file.: The value of *Angular_source* is used to indicate the source of the data for angular distribution of the incident atoms. If *Angular_source* = **1**, the incident angular distribution is defined by a function and a set of angles. In this case, the mean incident direction is defined by the pair θ_0 and φ_0 (one pair per incident atom type). If *Angular_source* = **0**, NASCAM reads the angular distribution from a file called *Angular_file*, and θ_0, φ_0 and $\Delta\theta$ (see next) as specified in the *input.txt* file are ignored. If *Angular_source* = **2**, a uniform angular distribution is used.

Theta0, phi0, dtheta (deg): for their definition see figures 14a and 14b. The angular distribution of the incident atoms around the mean incident direction is visualized as a cone like described in figures 16a and 16b.

Admitted values: θ_0 [0, 90°], φ_0 [-180°; 180°], $\Delta\theta$ [0, 90°].

During the simulation, the origin of the cone, i.e. the starting point of atoms being deposited, is chosen randomly over the defined sample surface. For fixed angle deposition, one should specify θ_0 and φ_0 and fix $\Delta\theta = 0°$ (*Angular_source* = **1)**.

Specie:Substrate: for the substrate it is necessary to indicate its name and atomic mass: *Specie:Substrate Cu 63.54*

Temp,(eV):

The next line of the *input.txt* file gives the possibility to establish the value of the simulation temperature. This is given in eV. Each atom is in local equilibrium with the substrate, and no temperature difference exists from site to site.

Save_data:

The first parameter on the *Save_data* line is used to switch on/off the recording of intermediate structures obtained during the current simulation in the *film_growth.xyz* file (see Section 5). If the first parameter is set to 1, then the output structure files will be updated every *N* deposited particles, *N* being the second parameter on the same line. If the first parameter is set to zero, no intermediate output files will be created.

If the number of atoms to deposit (*NDep*) is set to 0 (e.g. the user is studying the annealing effect), then output files will be created in time intervals equal to the time in which it would be necessary to deposit *N* equivalent particles ($N^* \tau$).

Surface_binding_energy_of_the_substrate,(eV),
surface_binding_energy_of_the_film,(eV):

These data are used to predict whether a defect is created on the substrate or an atom can be dislodged or sputtered from the film. As first approximation, the values can be the same as the heat of sublimation used to calculate the sputtering in SRIM code.Sputtering_Thresh/Yield 0.0 0.0

w_rot(1/s);init_tilt(deg);A_osc(deg);w_osc(1/s):

These fields are used to specify the parameters corresponding to the eventual movements of the substrate (or, equivalently, particle source) during the deposition process: rotation, initial tilt, oscillation amplitude and frequency (see the description before).

w_rot(deg/s); init_tilt(deg);A_osc(deg);w_osc(1/s) 0.0 0 0 4.5

Forced_deposition:

Sometimes the number of calculation steps between two successive deposition events is very large, and simulation takes forever. Usually this takes place at an elevated temperature when the activation energy for diffusion is low. Now it is possible to "boost" the simulations by setting the maximum number of steps between two successive deposition events. This can be regarded as increasing the deposition rate.

5.1.2 substrate.xyz

The *substrate.xyz* file contains information about the initial structure of the system. If, prior to the deposition, there are no gas or mobile particle and the substrate is flat and regular[2], this file may be absent from input folder. Its first line indicates the number of particles belonging to the initial system, and the second line is a comment. All subsequent lines consist of 4 columns *(element, x, y, z)*, first of them containing information about the chemical nature and the other three corresponding to the x, y and z coordinates (in lattice parameter units) of a particle belonging to the initial configuration of the system (Fig. 27).

A structure obtained in a previous simulation can be used as substrate or the user can edit its own *substrate.xyz* file. For the last option, details on the entire procedure are given below in the <<***Short guide "How to make a custom substrate file"***>> chapter**.** In both cases the x and y dimensions of the structure contained in the *substrate.txt* file must correspond to those defined in the *input.txt* file. If not, the program writes an error message in the *log_file.txt* file and stops.

[2] If the user wants to simulate the deposition on a substrate whose surface includes channels or holes, he may use the ancillary code Make Structure provided with this package (see section 12 and The Ancillary codes manual).

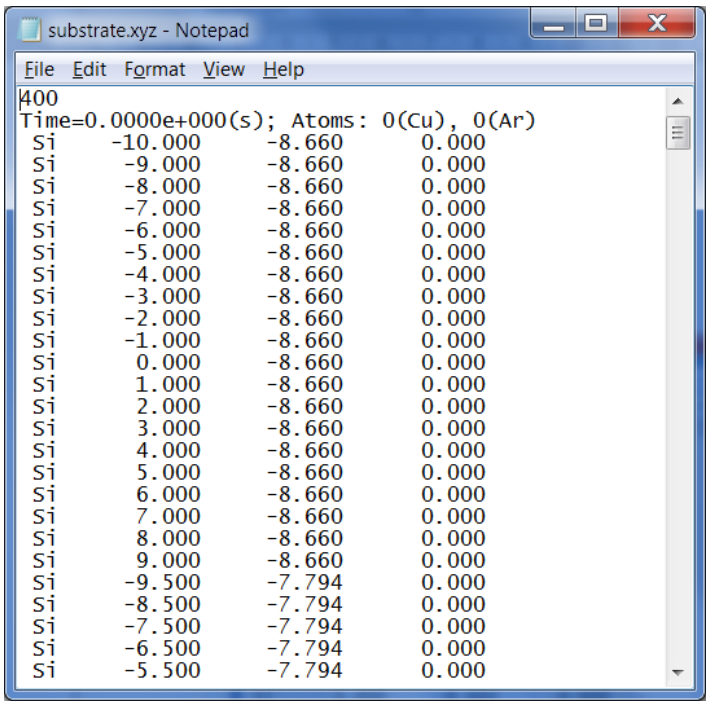


Figure 27. An example of substrate.xyz file.

If the *substrate.xyz* file is absent, it is assumed that the initial substrate is flat and has no defects. In this case NASCAM creates a new *substrate.xyz* file where all the sites at zero level are occupied by substrate atoms as defined in *input.txt* file.

NASCAM allows the deposition of multilayers. For example it is possible to deposit Cu on C, then Co atop, then again Cu etc. But when one user deposits Cu on the sandwich of C & Cu & Co, he should understand that only Cu atoms (newly deposited and previously deposited also) are mobile and all others atoms (C, and previously deposited Co atoms) are regarded as substrate atoms and can't move.

If *element* in *substrate.xyz* file coincides with one of the elements specified in *input.txt*, atom_name, gas_name or with defect name "Xx", then this particle is regarded as a pre-deposited mobile particle, gas particle or pre-existing defect respectively. In all other cases it is regarded as a substrate particle. There is only one exception, if the z-coordinate of a mobile (according to its chemical nature) particle equals to 0, then this particle is regarded as a substrate particle in what concerning its diffusion properties. It will never diffuse or evaporate but keeps its nature - this is natural as there are no mobile particles at zero level, only substrate, defects, or gas atoms.

In the current version of NASCAM all needed information about each kind of particle (substrate, gas and mobile particle), including its atomic mass and specific deposition parameters (for gas and mobile particles) are given on a corresponding single line in the *input.txt* file, as mentioned above. All particles in the *substrate.xyz* file, having a name different from mobile atom name or gas atom name defined in the *input.txt* file will be considered as substrate particles and they will not be taken into account in none of the thermally activated processes (diffusion and evaporation).

Let us consider the next definitions for the system symmetry, dimensions and species of the *input.txt* file:

```
Substrate_type      0
Dimensions          197   26   10   2000   0
…
Specie:Metal        1.0 Al 26.982 1 0.5 histo.txt 1 0.0 0.0 0.0 ang_dstr.txt
Specie:Gas          0.0 Ar 39.948 1 0.5 histo_gas.txt 1 0.0 0.0 0.0 ang_dstr_gas.txt
Specie:Substrate    Cu 63.54
…
```

and a hypothetic *substrate.xyz* file containing next lines:

```
7
hypothetic substrate.xyz file
Cu  -1   1   0
Cu   1   1   1
Ar   2   1   1
Al   2  -2   1
Al   3   3   0
Si   1   5   1
Xx  -4  -3   0
```

Firstly, as it was already mentioned, the first line of the *substrate.txt* file gives information regarding the number of particles in the system, while the second line is a comment line. All other lines contain information about the system structure, a line corresponding to a single particle definition: its nature and x, y, z coordinates.

In this example, both Cu particles will be considered as being substrate particles, the Ar particle as being a gas particle and the first Al particle as being a mobile particle. The second Al particle - because of its z coordinate, and the Si particle - because this kind of particle is not defined in the *input.txt* file, will be considered as being substrate particles. So, unlike the first Al particle, the second one will never participate to a thermal activated process. The last particle being a defect will never diffuse or evaporate (it is in fact a substrate particle).

Short guide "How to make a custom substrate file"

Sometimes it is necessary to make a deposition on a specific substrate. Such a substrate may contain some regular or irregular pre-existing structures. Regular structures may be built by a MakeSubstrate plugin (see the **Plugins Manual**). Irregular structures such an island or group of islands or one or several defects at the substrate a user should create by himself. Let us discuss how to create a custom substrate file using as an example a structure shown in figure 28. This structure has 5 atoms positioned at the layer with z=1.

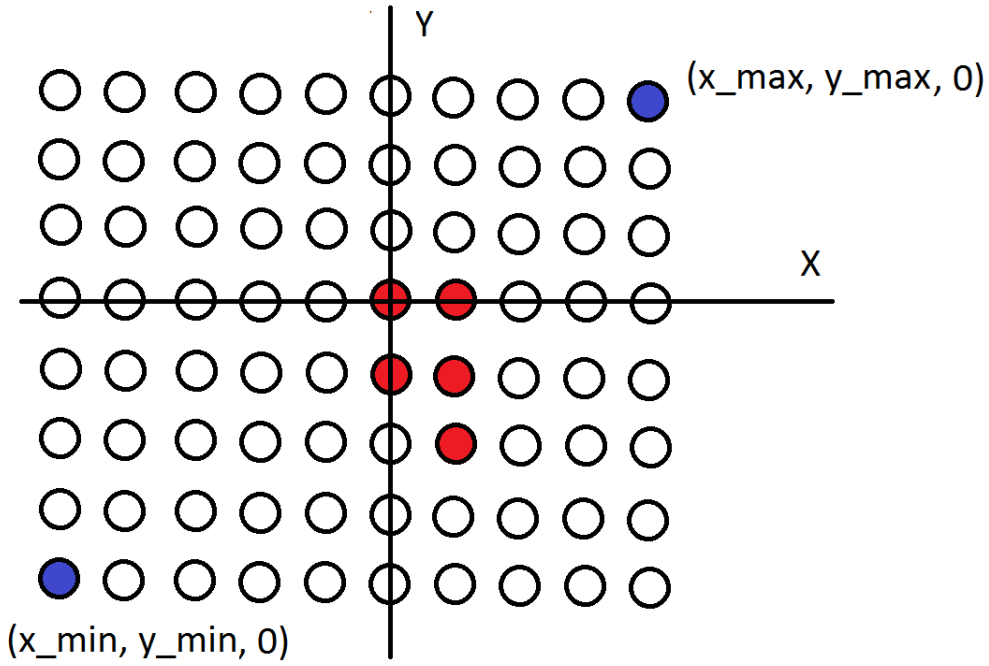


Figure 28. An illustration of a custom substrate file. The red circles correspond to atoms that form a required structure and the blue circles correspond to a nodes which define the substrate dimensions.

First of all, it is necessary to define a substrate type, cubic or hexagonal, and its dimensions, *NXmax* and *NYmax*. For this example substrate type is cubic, and *NXmax* = 10, *NYmax* = 8. According to what was written above one can calculate the coordinates of the nodes corresponding to the limits of the substrate:

*x_min = -0.5*NXmax = -5,* *y_min = -0.5*NYmax =-4,*
*x_max = 0.5*NXmax-1 = 4* *y_max = 0.5*NYmax-1 = 3.*

Remember that the same values for *NXmax* and *NYmax* must appear in *input.txt* file. If one changes these values in the *input.txt* file then it is necessary to re-calculate the limits in the *substrate.xyz* file. Note that the absolute value of *x_min* and *x_max*, *y_min* and *y_max* are not the same. This is due the periodical boundary conditions imposed to the substrate. Once the limits are calculated it is necessary to write the corresponding lines in *substrate.xyz* file:

El -5 -4 0
El 4 3 0

These two lines are obligatory!

Now it is necessary to put the coordinates of the atoms in *substrate.xyz* file. For the cubic substrate, *x* and *y* coordinates of the atoms are calculated according to the following expression:

$$x = i - \frac{NXmax}{2}, \quad y = j - \frac{NYmax}{2},$$

where *i* and *j* are integers varying in between 0 and *NXmax-1*, and 0 and *NYmax-1* respectively. For the given example the pairs *(i, j)* are (5,3), (5,4), (6,2), (6,3), (6,4). The *(i, j)* which

corresponds to *(x_min, y_min)* is (0, 0). As a result, the following lines appear in the substrate file:

```
Me      0.0     -1.0    1.0
Me      0.0      0.0    1.0
Me      1.0     -2.0    1.0
Me      1.0     -1.0    1.0
Me      1.0      0.0    1.0
```

Finally, it is necessary to add a JMOL header to the file:

```
Total number of atoms in the substrate file
A comment
```

The final result is:

```
7
Susbtrate file
El      -5      -4      0
El      4       3       0
Me      0.0     -1.0    1.0
Me      0.0      0.0    1.0
Me      1.0     -2.0    1.0
Me      1.0     -1.0    1.0
Me      1.0      0.0    1.0
```

If one wants to use a hexagonal structure then the limits of the substrate and coordinates of the atoms of the structure are calculated as follow:

$$x\_min = -\frac{NXmax}{2},\quad y\_min = \frac{\left(-\frac{NYmax}{2}\right)\sqrt{3}}{2},$$

$$x\_max = \frac{NXmax - 1}{2},\quad y\_max = \frac{\left(\frac{NYmax}{2} - 1\right)\sqrt{3}}{2},$$

$$x = i - \frac{NXmax}{2} + \frac{mod(j,2)}{2}\quad ,\quad y = \frac{\left(j - \frac{NYmax}{2}\right)\sqrt{3}}{2},$$

where *mod(j,2)* equals to 0 if *j* is even and *mod(j,2)* equals to 1 if *j* is odd.

If it is necessary to have a defect in the substrate, then one can put a “specie” called Xx in the substrate file. There is only one difference between a defect read from the *substrate.xyz* file and the one created during the current simulation: the first one is a stable defect while the second one is a reversible defect (see before).

Innovative Coating Solutions,
Place Saint-Pierre, 11, B-5380 Forville., Belgium
slu@incosol4u.com www.incosol4u.com

It is possible to resume a calculation using the results of a previous simulation as a starting point. To do that, simply rename the *coating.xyz* output file (for the description of *coating.xyz* see section 5.2.2) into *substrate.xyz* and use it as an input file.

5.1.3 Energy distribution data file

The **Energy_file** (e.g. en_distr.txt) contains two columns (Fig. 29). The first column is the energy *E(N)*, and the second is the energy distribution function *f(E)* i.e. relative number of atoms which have energy in the interval *E(N) – E(N+1)*.

The interval between subsequent values of energy may be arbitrary and may vary from line to line. For each energy interval NASCAM generates atoms with the energies evenly distributed from E(N) to E(N+1).

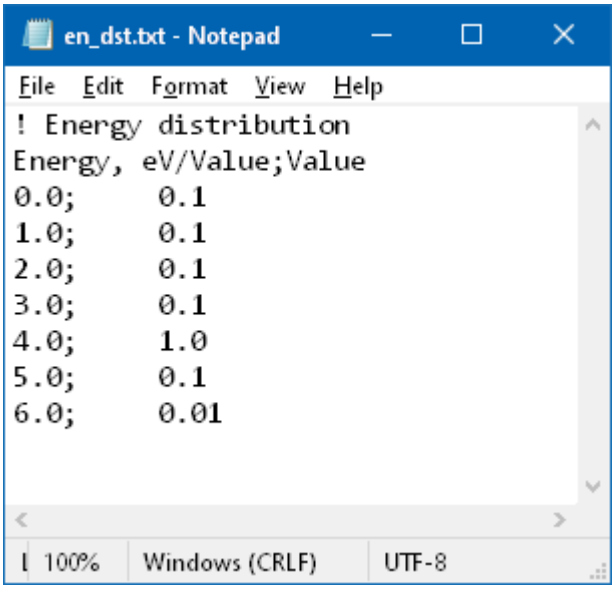


Figure 29. Example of energy distribution data file.

5.1.4 Angular distribution data file

The structure of this data file is similar to the structure of the energy distribution data file (Fig. 30). The first and the second columns are the angles θ , φ, respectively, and the third column contains the distribution probabilities. This file may be regarded as a tabulated representation of an angular distribution function $p=f(\theta, \varphi)$.

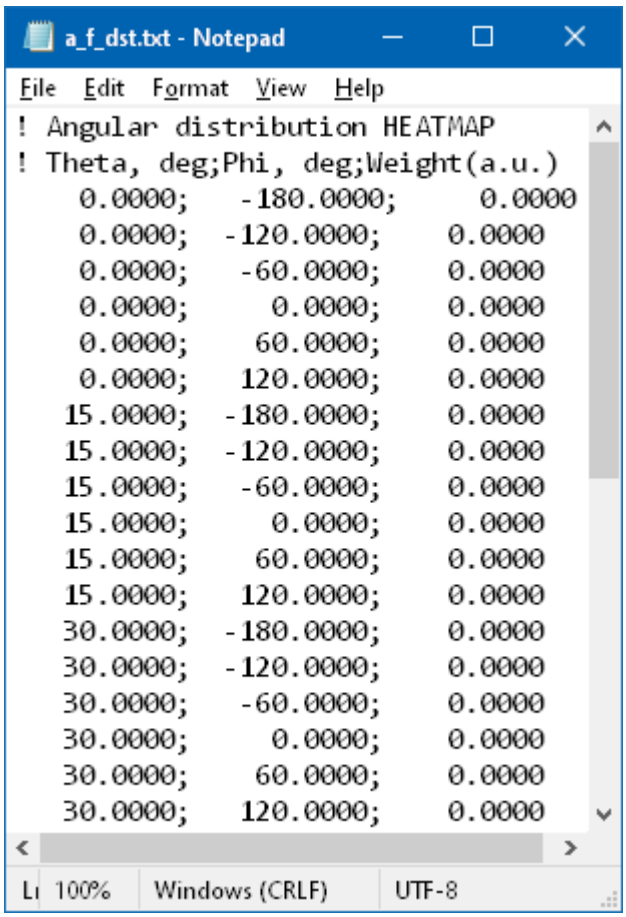


Figure 30. An example of angular distribution data file (from example 6).

The data must fill the regular grid for (θ, φ); θ varies in the range [0, 90°] and φ varies in the range [-180°, 180]. The grid step does not matter, but the structure of the file must be as on fig. 30. For each θ, φ is varied in the range [-180, 180], then θ is varied in the range [0, 90]. The upper limits of the ranges should not be included.

During the execution of NASCAM, the pair (θ, φ) which determines the incident direction of an atom is chosen stochastically but in accordance with the probabilities given in the fourth column.

For example, the distribution of θ, φ as specified from the file in figure 30 is shown in figure 31 (results of simulation of example 6).

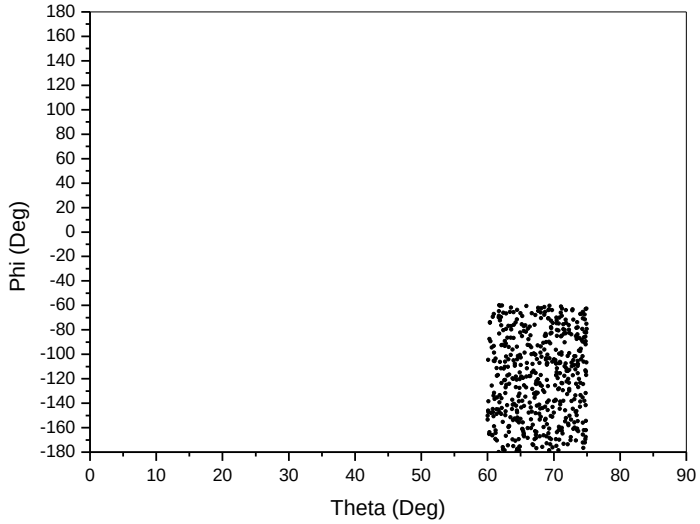


Figure 31. Chosen values of theta and phi as specified in the example of angular distribution data file in figure 30.

5.2 OUTPUT FILES

The resulted structures of the simulation are placed into two files: *coating.xyz* and *columns.xyz* that contains the information about the final configuration of the system. Files with additional helpful information are also created/updated (*film_growth.xyz, log_file.txt*…).

5.2.1 coating.xyz and columns.xyz

Both files contain the final (last, if intermediate results are generated) 3D-structure resulted from the current simulation. They have the same structure: the first line contains the total number of particles in the system, the second line is a comment line and all other following lines contain information regarding the nature of a given particle and its coordinates in lattice parameter units (particle_name coord_x coord_y coord_z). There are two additional columns that contain some auxiliary information which is necessary for multi-layer modeling. These files can be easily visualized using JMOL (see below). There is a difference between these two files: while the coating.xyz file just contains all particles belonging to the system (including the substrate particles and all particles deposited during a previous simulation) and contains information about composition of the coating, the columns.xyz presents the morphological structure of the coating.

At an early stage of deposition, when the coverage of the substrate is low, deposited atoms form islands or nucleation sites at the substrate. Each of these islands can give birth to a new column or new micro grain. During the deposition newly deposited atoms are attached to already existing columns or the can give birth to a new column. As the deposition process goes on the columns grow and their growth determine the structure of the coating. For the further information one can see Example 19. **Note, that columns.xyz file is created only when the option "Save intermediate 3D structures" is ticked.**

5.2.2 film_growth.xyz

If requested by setting the switch for *Save_data* (first number) to 1 in the *input.txt* file, a file called *film_growth.xyz*, is generated and then updated each *N* deposited particles, or "equivalent particles" during an annealing process, according to the second parameter (*N*) on the same *Save_data* line. This file contains the intermediate snapshots of the entire structure and may be used to observe its evolution in time.

Contrary to *coating.xyz* file that contains only the last saved snapshot of the system structure, the *film_growth.xyz* file contains all periodically saved structure snapshots (it is an appended file rather than an updated file).

5.2.3 log_file.txt

This file is an execution log and it is generated in the beginning of each simulation. It contains information about the used NASCAM version, the value of the random number generator seed, all input data read from the *input.txt* file, the computed event rates for all defined events and the characteristic time of the system (Fig. 32).

If the simulation stops due to an inadequate setting, the log_file.txt file will contain a corresponding error message. This file is updated each time when an important event happens, as for example a modification of the simulation box z-dimension.

It is also periodically updated, if this action is explicitly requested in the *input.txt* file. In this case, setting to *N* the first parameter on the first active line in the *input.txt* file (*Simulation_option*), NASCAM will update the *log_file.txt* file each *N* deposition events (or, "equivalent deposited particles" during an annealing process). Such update consists in a series of instantaneously collected/computed statistic information, information displayed simultaneously on the computer screen, too.

Some *.cvs* files will also be updated (appended), their update being controlled by the same *Simulation_option* first parameter.

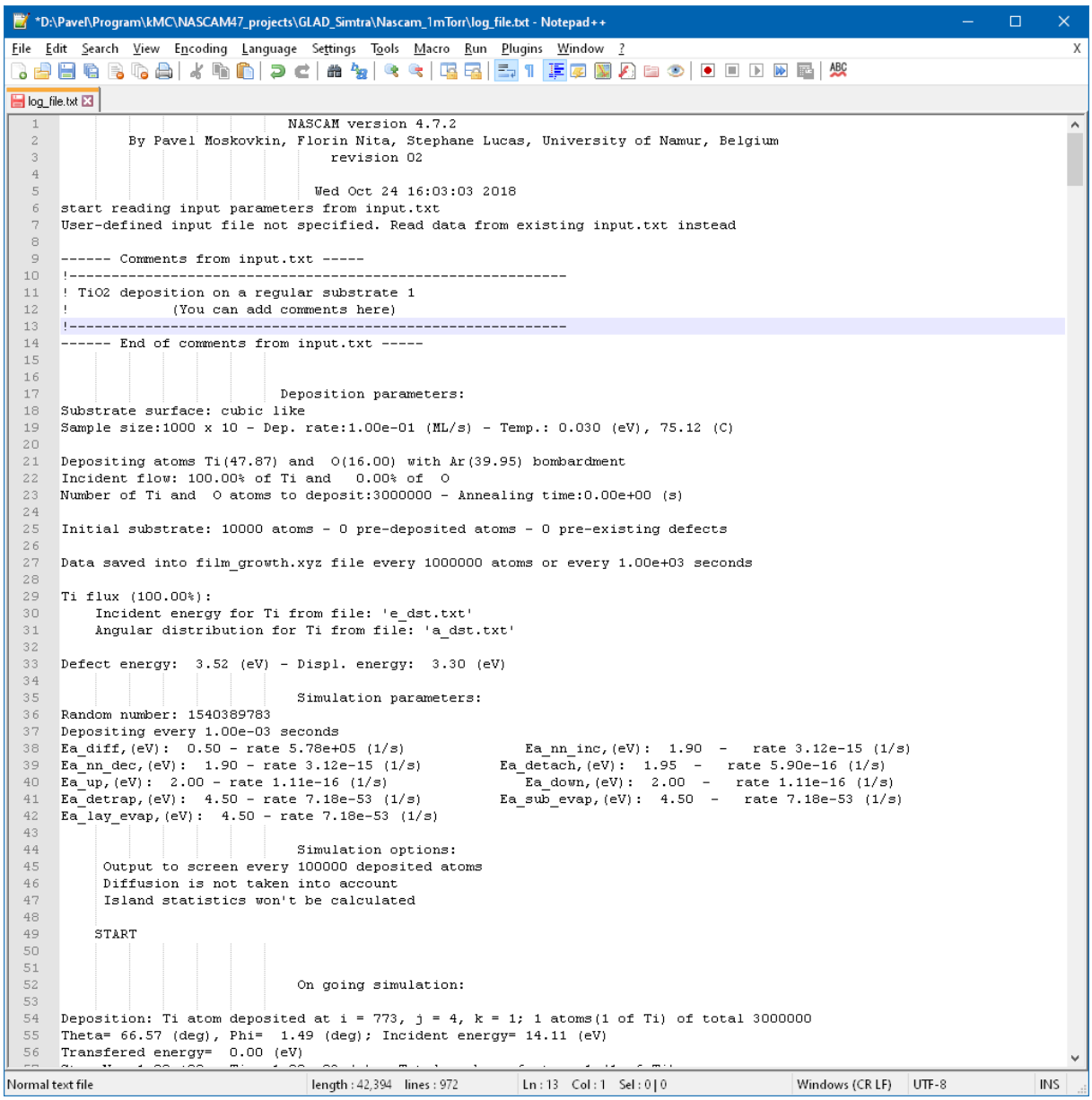
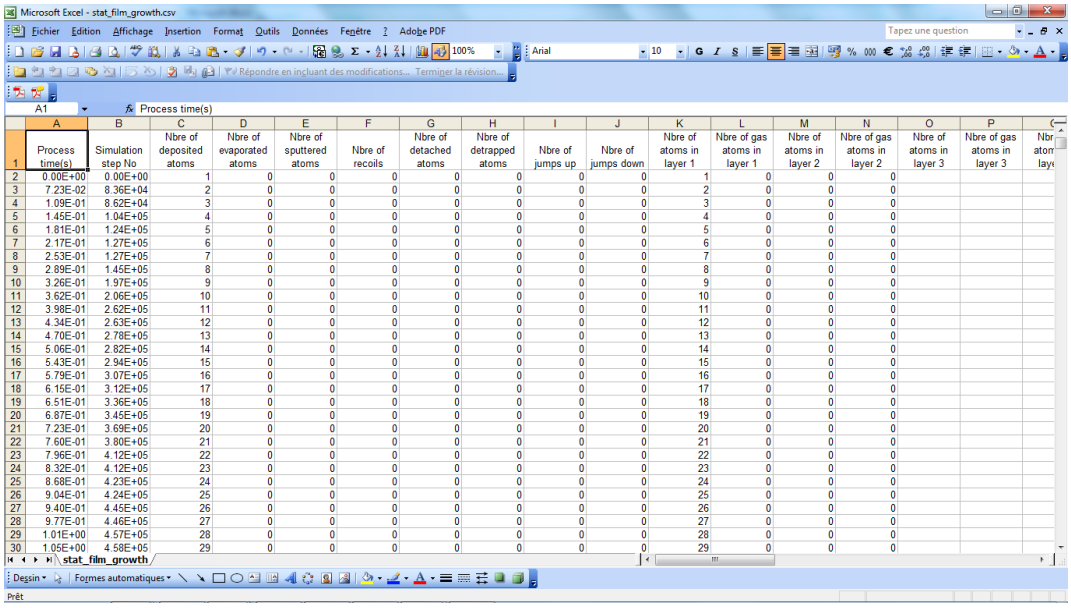


Figure 32. An example of *log_file.txt* file.

5.2.4 stat_film_growth.csv

This file contains information about the growth of the film (Fig. 33).

It is semicolon delimited and therefore can easily be read in a data manipulation software. Columns contain data about the number of events, such as the detachment of an adatom from an island, the de-trapping of an adatom etc.



Process time(s)	Simulation step No	Nbre of deposited atoms	Nbre of evaporated atoms	Nbre of sputtered atoms	Nbre of recoils	Nbre of detached atoms	Nbre of detrapped atoms	Nbre of jumps up	Nbre of jumps down	Nbre of atoms in layer 1	Nbre of gas atoms in layer 1	Nbre of atoms in layer 2	Nbre of gas atoms in layer 2	Nbre of atoms in layer 3	Nbre of gas atoms in layer 3
0.00E+00	0.00E+00	1	0	0	0	0	0	0	0	1	0	0	0		
7.23E-02	8.36E+04	2	0	0	0	0	0	0	0	2	0	0	0		
1.09E-01	8.62E+04	3	0	0	0	0	0	0	0	3	0	0	0		
1.45E-01	1.04E+05	4	0	0	0	0	0	0	0	4	0	0	0		
1.81E-01	1.24E+05	5	0	0	0	0	0	0	0	5	0	0	0		
2.17E-01	1.27E+05	6	0	0	0	0	0	0	0	6	0	0	0		
2.53E-01	1.27E+05	7	0	0	0	0	0	0	0	7	0	0	0		
2.89E-01	1.45E+05	8	0	0	0	0	0	0	0	8	0	0	0		
3.26E-01	1.97E+05	9	0	0	0	0	0	0	0	9	0	0	0		
3.62E-01	2.06E+05	10	0	0	0	0	0	0	0	10	0	0	0		
3.98E-01	2.62E+05	11	0	0	0	0	0	0	0	11	0	0	0		
4.34E-01	2.63E+05	12	0	0	0	0	0	0	0	12	0	0	0		
4.70E-01	2.78E+05	13	0	0	0	0	0	0	0	13	0	0	0		
5.06E-01	2.82E+05	14	0	0	0	0	0	0	0	14	0	0	0		
5.43E-01	2.94E+05	15	0	0	0	0	0	0	0	15	0	0	0		
5.79E-01	3.07E+05	16	0	0	0	0	0	0	0	16	0	0	0		
6.15E-01	3.12E+05	17	0	0	0	0	0	0	0	17	0	0	0		
6.51E-01	3.36E+05	18	0	0	0	0	0	0	0	18	0	0	0		
6.87E-01	3.45E+05	19	0	0	0	0	0	0	0	19	0	0	0		
7.23E-01	3.69E+05	20	0	0	0	0	0	0	0	20	0	0	0		
7.60E-01	3.80E+05	21	0	0	0	0	0	0	0	21	0	0	0		
7.96E-01	4.12E+05	22	0	0	0	0	0	0	0	22	0	0	0		
8.32E-01	4.12E+05	23	0	0	0	0	0	0	0	23	0	0	0		
8.68E-01	4.23E+05	24	0	0	0	0	0	0	0	24	0	0	0		
9.04E-01	4.24E+05	25	0	0	0	0	0	0	0	25	0	0	0		
9.40E-01	4.45E+05	26	0	0	0	0	0	0	0	26	0	0	0		
9.77E-01	4.46E+05	27	0	0	0	0	0	0	0	27	0	0	0		
1.01E+00	4.57E+05	28	0	0	0	0	0	0	0	28	0	0	0		
1.05E+00	4.58E+05	29	0	0	0	0	0	0	0	29	0	0	0		

Figure 33. An example of *stat_film_growth.csv* file in Excel™.

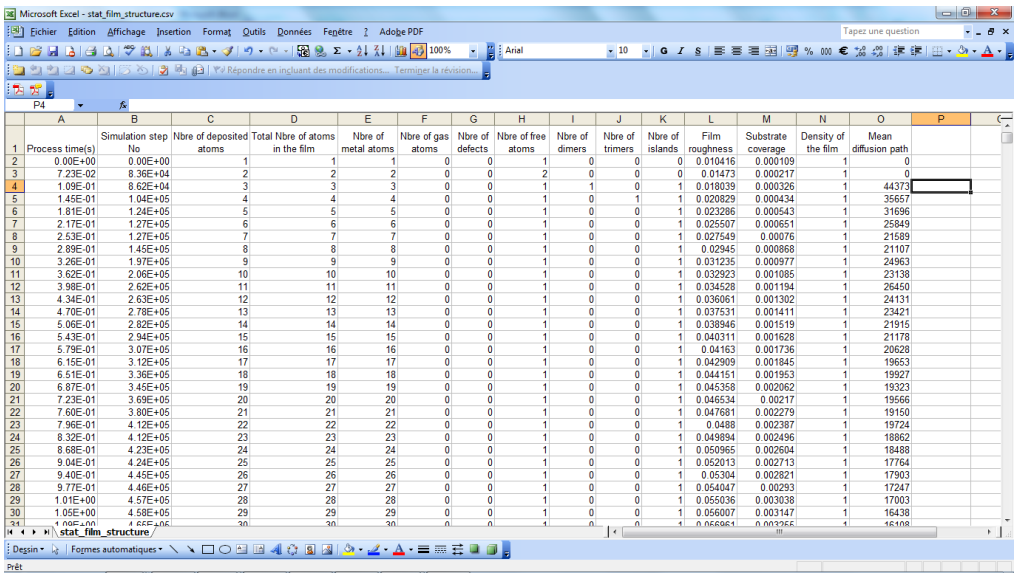
Also it contains information about the number of atoms at each mono-layer. We limit the information of atoms in various layers to 20 layers. The depth distribution of the various atoms can be plotted from the file *final_composition.csv*.

5.2.5 stat_film_structure.csv

This file contains a summary of the characteristics of the coating during the deposition (Fig. 34). The column headers are self-explanatory. It is semicolon delimited and therefore can easily be read in a data manipulation software.

Columns contain data about the number of atoms, defects etc. Other columns contain data about the number of dimmers, trimers, and the total number of islands, roughness, density, mean diffusion path.

Dimmers and trimers are adatom clusters, their adatoms being either nearest neighbours or next nearest neighbours one to each other (they can have different z coordinates).



Process time(s)	Simulation step No	Nbre of deposited atoms	Total Nbre of atoms in the film	Nbre of metal atoms	Nbre of gas atoms	Nbre of defects	Nbre of free atoms	Nbre of dimers	Nbre of trimers	Nbre of islands	Film roughness	Substrate coverage	Density of the film	Mean diffusion path
0.00E+00	0.00E+00	1	1	1	0	0	1	0	0	0	0.010416	0.000109	1	0
7.23E-02	8.36E+04	2	2	2	0	0	2	0	0	0	0.01473	0.000217	1	0
1.09E-01	8.62E+04	3	3	3	0	0	1	1	0	1	0.018039	0.000326	1	44373
1.45E-01	1.04E+05	4	4	4	0	0	1	0	1	1	0.020829	0.000434	1	35657
1.81E-01	1.24E+05	5	5	5	0	0	1	0	0	1	0.023286	0.000543	1	31696
2.17E-01	1.27E+05	6	6	6	0	0	1	0	0	1	0.025507	0.000651	1	25849
2.53E-01	1.27E+05	7	7	7	0	0	1	0	0	1	0.027549	0.00076	1	21589
2.89E-01	1.45E+05	8	8	8	0	0	1	0	0	1	0.02945	0.000868	1	21107
3.26E-01	1.97E+05	9	9	9	0	0	1	0	0	1	0.031235	0.000977	1	24963
3.62E-01	2.06E+05	10	10	10	0	0	1	0	0	1	0.032923	0.001085	1	23138
3.98E-01	2.62E+05	11	11	11	0	0	1	0	0	1	0.034528	0.001194	1	26450
4.34E-01	2.63E+05	12	12	12	0	0	1	0	0	1	0.036061	0.001302	1	24131
4.70E-01	2.78E+05	13	13	13	0	0	1	0	0	1	0.037531	0.001411	1	23421
5.06E-01	2.82E+05	14	14	14	0	0	1	0	0	1	0.038946	0.001519	1	21915
5.43E-01	2.94E+05	15	15	15	0	0	1	0	0	1	0.040311	0.001628	1	21178
5.79E-01	3.07E+05	16	16	16	0	0	1	0	0	1	0.04163	0.001736	1	20628
6.15E-01	3.12E+05	17	17	17	0	0	1	0	0	1	0.042909	0.001845	1	19653
6.51E-01	3.36E+05	18	18	18	0	0	1	0	0	1	0.044151	0.001953	1	19927
6.87E-01	3.45E+05	19	19	19	0	0	1	0	0	1	0.045358	0.002062	1	19323
7.23E-01	3.69E+05	20	20	20	0	0	1	0	0	1	0.046534	0.00217	1	19566
7.60E-01	3.80E+05	21	21	21	0	0	1	0	0	1	0.047681	0.002279	1	19150
7.96E-01	4.12E+05	22	22	22	0	0	1	0	0	1	0.0488	0.002387	1	19724
8.32E-01	4.12E+05	23	23	23	0	0	1	0	0	1	0.049894	0.002496	1	18862
8.68E-01	4.23E+05	24	24	24	0	0	1	0	0	1	0.050965	0.002604	1	18488
9.04E-01	4.24E+05	25	25	25	0	0	1	0	0	1	0.052013	0.002713	1	17764
9.40E-01	4.45E+05	26	26	26	0	0	1	0	0	1	0.05304	0.002821	1	17903
9.77E-01	4.46E+05	27	27	27	0	0	1	0	0	1	0.054047	0.00293	1	17247
1.01E+00	4.57E+05	28	28	28	0	0	1	0	0	1	0.055036	0.003038	1	17003
1.05E+00	4.58E+05	29	29	29	0	0	1	0	0	1	0.056007	0.003147	1	16438

Figure 34. An example of *stat_film_structure.csv* file in Excel™.

Roughness is defined as the root mean square of the height standard deviation (rms-roughness):

$$w = \sqrt{\langle h^2 \rangle - \langle h \rangle^2}, \qquad (22)$$

where *h(x,y)* is the local thickness of the film, and < > means the averaging over the whole substrate surface (*NXmax* × *NYmax*). Roughness is given in lattice parameter units.

The mean diffusion path is calculated as follows. For each atom the number of jumps between the deposition and the first attachment and between a possible next detachments and further attachments is calculated. Then this number id divided by the number of periods of time when the atom is free. Obviously the atom is free before the first attachment and between further detachments and subsequent attachments. In this way a mean diffusion path for a given atom can be estimated. After all this value is averaged over all atoms which are currently in the system.

The density of the film is calculated as the ratio of the number of deposited particles existing currently in the system and the number of lattice sites below the surface envelope, as shown in figure 35. In this example the density is equal to 34/42 = 0.81.

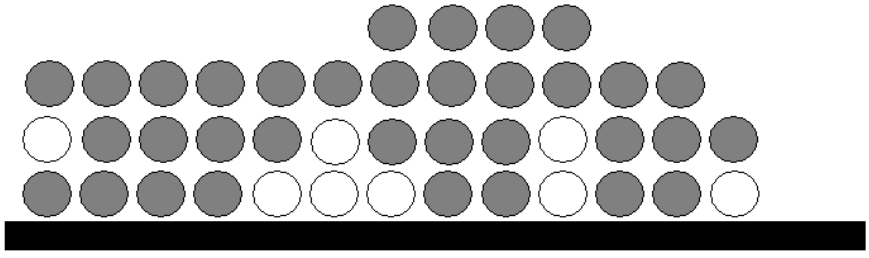
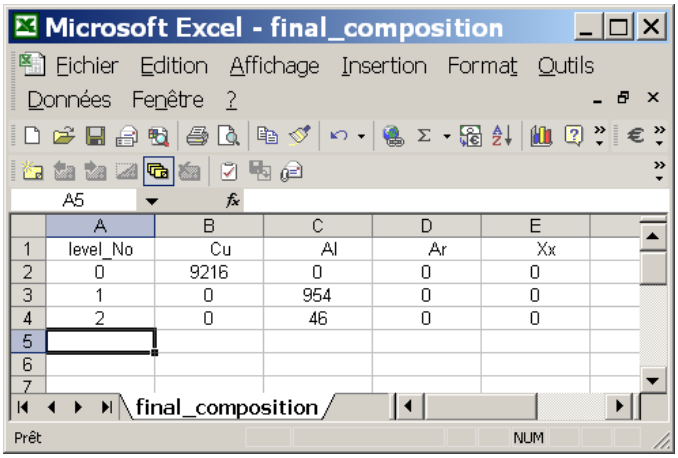


Figure 35. Lateral view of a system structure, its corresponding density being ~ 0.81. The grey circles represent occupied, while the white circles represent empty lattice sites.

5.2.6 final_composition.csv

This file can be used to plot the elemental composition versus depth. It contains the number of atoms of each species (including the defects) versus the level number (Fig. 36).

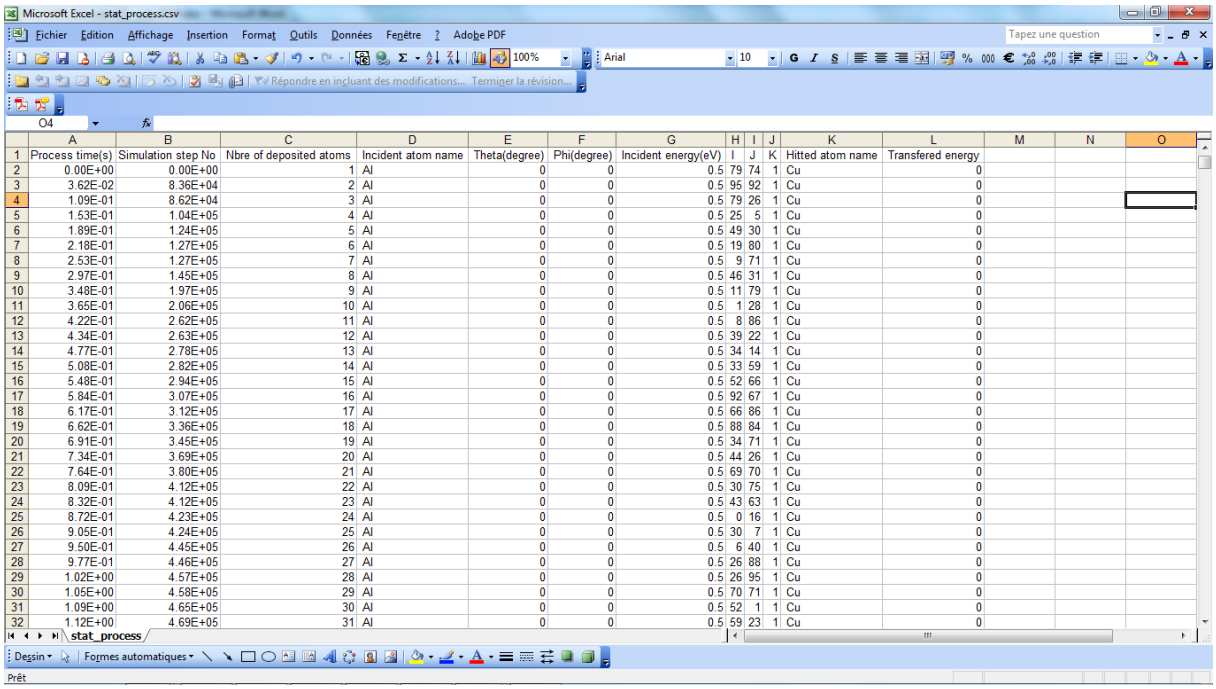


	A	B	C	D	E
1	level_No	Cu	Al	Ar	Xx
2	0	9216	0	0	0
3	1	0	954	0	0
4	2	0	46	0	0
5					
6					
7					

Figure 36. An example of *final_composition.csv* in Excel™.

5.2.7 stat_process.csv

This file contains a summary of all the events that happened during the deposition (Fig. 37).



	A	B	C	D	E	F	G	H	I	J	K	L
1	Process time(s)	Simulation step No	Nbre of deposited atoms	Incident atom name	Theta(degree)	Phi(degree)	Incident energy(eV)	I	J	K	Hitted atom name	Transfered energy
2	0.00E+00	0.00E+00	1	Al	0	0	0.5	79	74	1	Cu	0
3	3.62E-02	8.36E+04	2	Al	0	0	0.5	95	92	1	Cu	0
4	1.09E-01	8.62E+04	3	Al	0	0	0.5	79	26	1	Cu	0
5	1.53E-01	1.04E+05	4	Al	0	0	0.5	25	5	1	Cu	0
6	1.89E-01	1.24E+05	5	Al	0	0	0.5	49	30	1	Cu	0
7	2.18E-01	1.27E+05	6	Al	0	0	0.5	19	80	1	Cu	0
8	2.53E-01	1.27E+05	7	Al	0	0	0.5	9	71	1	Cu	0
9	2.97E-01	1.45E+05	8	Al	0	0	0.5	46	31	1	Cu	0
10	3.48E-01	1.97E+05	9	Al	0	0	0.5	11	79	1	Cu	0
11	3.65E-01	2.06E+05	10	Al	0	0	0.5	1	28	1	Cu	0
12	4.22E-01	2.62E+05	11	Al	0	0	0.5	8	86	1	Cu	0
13	4.34E-01	2.63E+05	12	Al	0	0	0.5	39	22	1	Cu	0
14	4.77E-01	2.78E+05	13	Al	0	0	0.5	34	14	1	Cu	0
15	5.08E-01	2.82E+05	14	Al	0	0	0.5	33	59	1	Cu	0
16	5.48E-01	2.94E+05	15	Al	0	0	0.5	52	66	1	Cu	0
17	5.84E-01	3.07E+05	16	Al	0	0	0.5	92	67	1	Cu	0
18	6.17E-01	3.12E+05	17	Al	0	0	0.5	66	86	1	Cu	0
19	6.62E-01	3.36E+05	18	Al	0	0	0.5	88	84	1	Cu	0
20	6.91E-01	3.45E+05	19	Al	0	0	0.5	34	71	1	Cu	0
21	7.34E-01	3.69E+05	20	Al	0	0	0.5	44	26	1	Cu	0
22	7.64E-01	3.80E+05	21	Al	0	0	0.5	69	70	1	Cu	0
23	8.09E-01	4.12E+05	22	Al	0	0	0.5	30	75	1	Cu	0
24	8.32E-01	4.12E+05	23	Al	0	0	0.5	43	63	1	Cu	0
25	8.72E-01	4.23E+05	24	Al	0	0	0.5	0	16	1	Cu	0
26	9.05E-01	4.24E+05	25	Al	0	0	0.5	30	7	1	Cu	0
27	9.50E-01	4.45E+05	26	Al	0	0	0.5	6	40	1	Cu	0
28	9.77E-01	4.46E+05	27	Al	0	0	0.5	26	88	1	Cu	0
29	1.02E+00	4.57E+05	28	Al	0	0	0.5	26	95	1	Cu	0
30	1.05E+00	4.58E+05	29	Al	0	0	0.5	70	71	1	Cu	0
31	1.09E+00	4.65E+05	30	Al	0	0	0.5	52	1	1	Cu	0
32	1.12E+00	4.69E+05	31	Al	0	0	0.5	59	23	1	Cu	0

Figure 37. An example of *stat_process.csv* in Excel™.

Columns contain data about the deposition process: incident angle, incident energy, substrate coordinate that the incident atom collided, type of hit atom & transferred energy.

5.2.8 island_distribution.csv

This file contains a histogram of the number of islands and their respective number of atoms (Fig. 38).

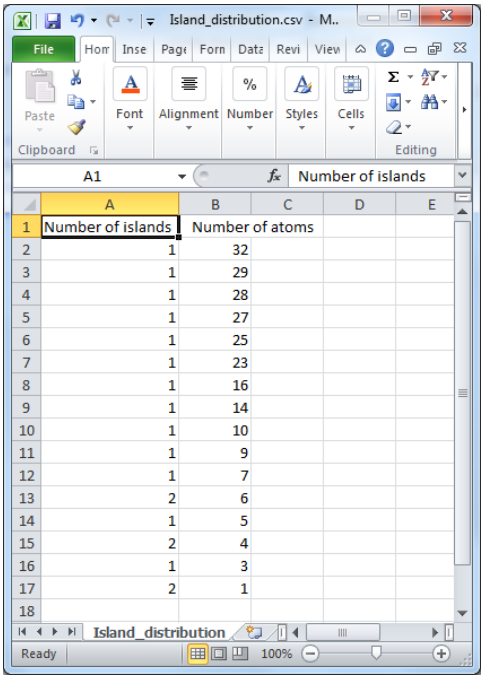


Figure 38. An example of *island_distribution.csv* file in Excel™.

Innovative Coating Solutions,
Place Saint-Pierre, 11, B-5380 Forville., Belgium
slu@incosol4u.com www.incosol4u.com

6 Execution of the code

First, all the input files (*input.txt*, *substrate.xyz*, ….) and the file **Nascam.exe** have to be placed into the same folder.

To execute the code, there are two options:

- Double click **Nascam.exe**.
- Use switches at the command lines (arbitrary order) to setup options:
  - –n filename. Specify a file which will used instead of *input.txt*
  - –ver. Check the version of the code without doing any simulations.

We recommend creating a new folder every time one executes the code with new input data, otherwise the old output data will be overwritten by the new outputs files.

We suggest firstly running the code with a set of parameters (NXmax and NYmax, deposition rate, and temperature) close to the real experimental/simulation conditions, but to set first the number of deposited atoms or the annealing time to low values (e.g. 100 atoms). Run the code and estimate the execution time. If the execution time is too long, edit the input data to have a reasonable execution time.

Innovative Coating Solutions,
Place Saint-Pierre, 11, B-5380 Forville., Belgium
slu@incosol4u.com www.incosol4u.com

7 JMOL plotting software

7.1 INTRODUCTION

NASCAM outputs ascii file having the suffix_jmol (XYZ) are compatible with JMOL (http://jmol.sourceforge.net/).

JMOL is an open-source Java viewer for three-dimensional chemical structures, with features for chemicals, crystals, materials and biomolecules. Features include the reading a variety of file types and the animation of multi-frame files (http://wiki.jmol.org/index.php/Main_Page).

We choose the simplest data file format for *substrate.xyz*, *coating.xyz*, …: XYZ format. It only contains atom coordinates in a Cartesian coordinate system (no bonds). It supports multi-model data (multi-frame, animations).

Figure 39 shows an example of an XYZ file:

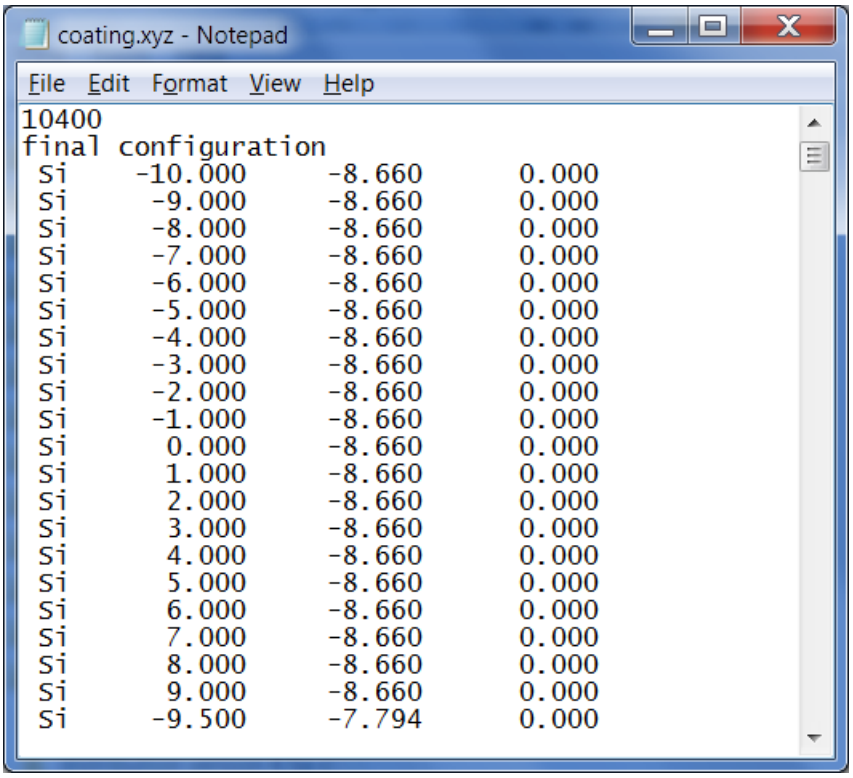


Figure 39. An example of *coating.xyz* file.

The first line contains the total number of atoms, the second line is a comment line, and then atoms are listed as: El X Y Z.

The XYZ format supports **multi-step datasets**. **Each dataset** is represented by a two-line "header," followed by one line for each atom. NASCAM generates an *film_growth.xyz* file that contains “snapshots” of the simulation saved according to the *Save_data* switch in the *input.txt* file.

Once NASCAM simulation is over and JMOL is properly installed, the user can drag and drop either *film_growth.xyz*, *substrate.xyz*, *coating.xyz* or *deposited_layer.xyz* to the JMOL window. The simulation results will be displayed immediately.

When multiple datasets are loaded, all the datasets are stored in the frame vectors within JMOL. They can be displayed one by one, in sequence or in animation. The user can then review the evolution of the simulation by viewing the snapshots in sequence.

7.2 JMOL BASIC

First read the manual for the mouse:

- Start JMOL
- Right click in the window. A menu appears
- Select “About ... JMOL xxxx … Mouse Manual”

Operations to load one single dataset (e.g. *coating.xyz*)

- Start JMOL and load a NASCAM *coating.xyz* file.
- Select “Display Bond None”
- Rotate the 3D model with the mouse,
- …

Operations to load multiple dataset (e.g. *film_growth.xyz*)

- Start JMOL and load a NASCAM *film_growth.xyz* file that contains multiple dataset (see for example, Example 2 (Ostwald ripening))
- Select “Display Bond None”
- Rotate the 3D model with the mouse.
- Press the “Right arrow” (go to next frame) on the upper button bar to select the next dataset. Look at the bottom bar, at the left: JMOL indicates that you have selected the second dataset.
- Rotate the 3D model with the mouse.
- …

If one wants to display every dataset in sequence, like a movie, select “Tools Animate Once” or “Tools Animate Loop” in the menus.

Innovative Coating Solutions,
Place Saint-Pierre, 11, B-5380 Forville., Belgium
slu@incosol4u.com www.incosol4u.com

8 Typical input data ranges

- Activation energies: 0 to 4.5 eV. Any value above 4 eV means that the corresponding physical event is switched off (look at the rate of the event in the *log_file.txt* file). Please note that this does not prevent the code to calculate the event probabilities. If one wants to really switch off diffusion, use the command line switches. This speeds up the code execution.
- Max number of layers (Zmax): up to the physical memory of your computer. It is not an essential variable and larger its value larger the amount of resources (memory), so keep it as small as possible having in mind that its value is automatically updated (increased) if necessary during a simulation.
- Energy distribution selection: 0 = read from file, 1 = use fixed energy value, 2 = computed using a built-in function
- Angular distribution selection: 0 = read from file, 1 = computed using a built-in function
- θ_0:[0° ÷ 90°], φ_0:[-180° ÷ 180], $\Delta\theta$:[0° ÷ 90°].

9 NASCAM version history

Version 1. First stable version. Main diffusion effects are taken into account.

Version 2. Energy of incident atoms is taken into account. Possibility to save intermediate results in **outN.txt** files is added. Statistics on the events is now gathered in a separate file. Format of output files **fbox.txt** and **density.txt** is slightly changed.

Version 3.

- 3.2: add substrate roughness (ancillary code Mk_strct) and allow the user to specify incident angles of the incoming species.
- 3.5dx: add evaporation, statistic2.csv, calculation of roughness, density, diffusion path, number of free atoms, dimmers, trimers, …

Version 4.3D

- Improve the energy transfer module
- Calculate substrate coverage
- Correct some minor bugs
- Add boosted mode
- New file format compatible with Jmol
- Add simultaneous rare gas bombardment
- Remove unnecessary calculations (deep atoms are prevented to move).
- Add dynamic memory allocation: limitation is due to computer physical memory
- Results rearranged in various csv files (film_statistics, final_composition, process_statistics, …).

Version 4.5 (release july 2014)

- Add deposition through a moving mask
- Add samples (or source) rotation and tilting capabilities
- Add command lines to prevent diffusion calculation and shorten output csv files
- Push the code to handle millions of atoms.

Version 4.5.1 (release september 2016)

- Sorting of the events has been optimized. Now only atoms belonging to the neighbourhood of the moving atom are evaluated.
- Dynamic allocation of the height Zmax of the simulation box
  Change the output: remove obsolete files ibox.txt, fbox.txt, outN.txt. Uses only files in JMOL format (XYZ)

Version 4.6.1 (release september 2017)

- General improvements and optimizations:

- Change the random number generator. For more details on this item see https://isocpp.org/files/papers/n3551.pdf.
- The output frequency during the annealing process is now better computed and the output .cvs files are generated according to the same rule as for the deposition process
- Fix a bug connected to the forced deposition option.
- Change the code architecture defining and adding new variables (nearest neighbours of a given lattice site and their corresponding number and list, surface envelope, the highest surface position…).
- Remove not necessary computation steps and variables considered obsolete
- Based on new introduced variables almost all functional procedure have been optimised, the most important of them being the update procedure: a local update is done after each modification of the system configuration for all direct and indirect affected particles. Such modification includes not only the main thermal activated event but also all connected events

induced by this one: diffusion and evaporation can be associated with gas desorption and deposition can be associated with sputtering and gas desorption as well.

- Degassing is done also locally and immediately after each main thermal activated event (deposition with or without sputtering/diffusion/evaporation); only after that, the list of particle to update is created.
- In deposition:
  - The trajectory of the incoming particle is computed not from the upper surface of the simulation box but from few layers above the maximum of the surface envelope of the already deposited structure inside the simulation box.
  - Fix a bug in choosing the coordinates of the initial attachment position.
  - Adding, in the case of cubic symmetry, the missing four next nearest neighbour positions to the lists of the initial and final positions for a recoil particle.
  - Fixed two bugs for the attachment conditions corresponding to a gas particle (threshold energy and the lateral nearest neighbours of the final attachment position)
  - Fix the error in computing the final position for a recoil particle by using the periodic boundary conditions.
  - Fix the bug in jump down procedure - error in choosing the final attachment position for a mobile particle.
- In diffusion:
  - Now a diffusing particle will never leave behind it a 3d isolated particle ("levitating particle").
  - Fixed an error in computing the number of processes associated to a single moving up/down diffusion event.
- In evaporation, as in deposition before, the trajectory, and its neighbourhood, of an evaporating particle is checked from the particle initial position not till the upper surface of the simulation box but till a z coordinate corresponding to the maximum of the surface envelope.
- System structure analysis procedures have been improved as well.

Version 4.7.2 (release September 2019)

- Reactive deposition has been added
- Major improvement of the code.
- Energy transfer model was improved.
- Several new examples were added.
- New plugins were added

Version 4.8.5 (release September 2024)

- MakeProcess "digital twin" tool was added
- New plugins were added.
- Column detection possibility was added.
- Format of substrate.xyz and coating.xyz are changed:
  - added layer index and number of atoms per cell to parameters of each atom,
  - comment line are extended.

10 References

1. Wang Z., Li Y and Adams J.B. Surf Sci. 450 (2000) 51; Wei Helin et al, Vacuum 52 (1999) 435.
2. C.H. Claassens et al, Journal of Physics: Conference Series 29 (2006) 185.
3. X. Tan, Y.C.Zhou, X.J.Zheng, Surface and Coating Technology 197 (2005) 288-293.
4. A.A. Schmidt, H. Eggers, K. Herwig, R. Anton, Surface Science, 349 (1996), 301.
5. O. Trushin, A. Karim, A. Kara, and T. S. Rahman, Phys. Rev. B 72, (2005), 115401.
6.W. Möller, Fundamentals of Ion-Surface Interaction, Short Resume of a lecture held at the Technical University of Dresden, Issue: Winter 2003/2004.
7. SImulation of Metal TRAnsport, available at http://www.draft.ugent.be/.

11 Citation

The following references are related to simulation performed by the authors using NASCAM.

It is kindly asked that users who publish results obtained from NASCAM cite one or more of the following articles:

- *"Simulation at high temperature of atomic deposition, islands coalescence, Ostwald and inverse Ostwald ripening with a general simple kinetic Monte Carlo code"*, S. Lucas, P. Moskovkin, Thin Solids Films (2010), Volume 518, Issue 18, 1 July 2010, Pages 5355-5361.
- *"Surface phenomena involved in the formation of Co nanoparticles on amorphous carbon and SiO2 deposited by magnetron sputtering"*, S. Lucas, J.F. Colomer, C. Bittencourt, P. Moskovkin, N. Moreau, Applied Physics A: Materials Science & Processing, Volume 99, Number 1 / 2010, 125-138.
- *"Computer simulations of the early stage growth of Ge clusters at elevated temperatures, on patterned Si substrate using the kinetic Monte Carlo method.",* P. Moskovkin and S. Lucas, Thin Solids Film, Volume 536, 1 / June 2013, 313–317.
- *"On the formation of the porous structure in nanostructured a-Si coatings deposited by dc magnetron sputtering at oblique angles"*, G. Vanda, P. Moskovkin, R. Alvarez, J. Caballero-Hernandez, R. Schierholz, B. Bera, J. Demarche, A. Palmero, A. Fernandez, S. Lucas, accepted for publication in NANO-104172 (2014).
- *"Synthesis of nanostructured Ti thin films by combining glancing angle deposition and magnetron sputtering: a joint experimental and modeling study"*,J.Dervaux, P.-A.Cormier, P.Moskovkin, O.Douheret, S.Konstantinidis, R.Lazzaroni, S.Lucas, R.Snyders, Thin Solid Films, Volume 636, 2017, 644-657.
- *"TiOx deposited by magnetron sputtering: a joint modelling and experimental study"*, R. Tonneau, P. Moskovkin, A. Pflug, S. Lucas, J. Phys. D. Appl. Phys. 51 (2018) 195202, https://doi.org/10.1088/1361-6463/aabb72 *"Wide range investigation of duty cycle and frequency effects on bipolar magnetron sputtering of chromium nitride"*, E. Haye, J.L. Colaux, P. Moskovkin, J.-J. Pireaux, S. Lucas, *Surface Coating and Technology 350 (2018) 84-94*, https://doi.org/10.1016/j.surfcoat.2018.07.009

- *"Metal filling by high power impulse magnetron sputtering"*, L. Jablonka, P. Moskovkin, Z. Zhang, S.-L. Zhang, S. Lucas, T. Kubart, *J. Phys. D: Appl. Phys.* 52 (2019) 365202, https://doi.org/10.1088/1361-6463/ab28e2
- "Correlation of structural and optical propertiesusing virtual materials analysis", H. Badorreck, M. Steinecke, L. Jensen, D. Ristau, M. Jupé, J. Müller, R. Tonneau, P. Moskovkin, S. Lucas, A. Pflug, L. Grinevičciūtė, A. Selskis, T Tolenis, Optics Express **27**(16) (2019), 22209-22225, https://doi.org/10.1364/OE.27.022209

12 Annex 1: Another description of the input.txt file

Table A1.1. Short description of input.txt file content.

Names of fields in input.txt file	Typical values for corresponding variables / Names of input variables (their meaning is given in the Table 2)
Simulation_options	1000 1 0 / statCmpSw, diffCmpSw, islandCmpSw
Substrate_type	0 / symm
Dimensions	3 3 10 100000 0 / NXmax, NYmax, NZmax, NDep, Tmax
Deposition_rate,(ML/s)	0.05 / F
Prefactor_correction	1.0 / w_0Corr
Masked_deposition	0 / maskDepSw
Pattern_name	LARN.txt / pattFile
Source_linear_movement	0 0.0 / maskMvtSw, V
Ea_diff,(eV)	0.9 / Ea_diff
Ea_nn_inc,(eV)	0.9 / Ea_nn_inc
Ea_nn_dec,(eV)	0.9 / Ea_nn_dec
Ea_detach,(eV)	1.0 / Ea_detach
Ea_up,(eV)	1.1 / Ea_up
Ea_down,(eV)	1.1 / Ea_down
Ea_detrap,(eV)	4.5 / Ea_detrap
Ea_sub_evap,(eV)	4.5 / Ea_sub_evap
Ea_lay_evap,(eV)	4.5 / Ea_lay_evap
Specie:Metal	1.0 Co 63.546 1 0.029 eDist.txt 1 80.0 0.0 1.0 aMDist.txt / *
Specie:Gas	2.0 Ar 39.948 1 0.029 histoGas.txt 1 0.0 0.0 0.0 aGDist.txt / **
Specie:Substrate	Fe 28.085 / elemS, massS
Temp,(eV)	0.035 / T
Save_data	1 10000 / saveDataSw, saveDataNo
Surface_binding_energy _of_the_substrate,(eV)	3.0 / subBindEn
Surface_binding_energy _of_the_film,(eV)	3.0 / layBindEn
Sputtering_Thresh/Yield	0.0 0.0 !!! NOT YET IMPLEMENTED
w_rot(1/s);init_tilt(deg); A_osc(deg);w_osc(1/s)	0.0 0.0 0.0 0.0 / w_rot, Ω_0, A, w_osc
Forced_deposition	1.0E15 / maxDiffNo
*Specie:Metal_1/2	concM_1/2, elemM_1/2, massM_1/2, enMSw_1/2, enAvM_1/2, enFileM_1/2, angMSw_1/2, θ_0_M_1/2, φ_0_M_1/2, $\Delta\theta$_M_1/2, angFileM_1/2
*Specie:Reactive_1/2	concR_1/2, eleRR_1/2, RassR_1/2, enRSw_1/2, enAvR_1/2, enFileR_1/2, angRSw_1/2, θ_0_R_1/2, φ_0_R_1/2, $\Delta\theta$_R_1/2, angFileR_1/2
**Specie:Gas	concG, elemG, massG, enGSw, enAvG, enFileG, angGSw, θ_0_G, φ_0_G, $\Delta\theta$_G, angFileG

Table A1.2. The names of variables used by NASCAM and their meaning.

Names of input variables	Meaning
statCmpSw	statistics computation switch: 0 – Not / larger than 0 – Yes, each "statCmpSw" deposition events
diffCmpSw	diffusion computation switch: 0 – Not / 1 – Yes
islandCmpSw	island density computation switch: 0 – Not / 1 – Yes
symm	substrate symmetry switch: 0 – cubic / 1 – hexagonal
Nxmax, Nymax, Nzmax	system dimensions on x, y and z [lattice parameter units]
Ndep	number of atoms to deposit during current simulation,
Tmax	annealing time [s]
F	deposition rate in Mono-Layer/Seconds (ML/s)
w_0Corr	pre-exponential factor correction:1 for particle diffusion / less than 1 for cluster diffusion
maskDepSw	masked deposition switch: 0 – no mask / 1 – there is a mask
pattFile	the name of the file containing the mask definition (e.g. pattern.txt)
maskMvtSw	source linear movement switch: 0 – no movement / 1 – there is a movement
V	mask/source speed [lattice parameters/sec]
Ea_diff	activation energy for the isolated particle in-layer diffusion – such event corresponds to $nn_i = 0$ * [eV]
Ea_nn_inc	activation energy for the particle in-layer diffusion with $nn_f \geq nn_i$ * [eV]
Ea_nn_dec	activation energy for the particle in-layer diffusion with $nn_f < nn_i$ * [eV]
Ea_detach	activation energy for the particle in-layer detachment diffusion – such event corresponds to $nn_f > 0$ * [eV]
Ea_up	activation energy for the particle layer-to-layer up diffusion [eV]
Ea_down	activation energy for the particle layer-to-layer down diffusion [eV]
Ea_detrap	activation energy for the particle detrapment [eV]
Ea_sub_evap	activation energy for the particle substrate-evaporation [eV]
Ea_lay_evap	activation energy for the particle layer-evaporation [eV]
elemM, elemR, elemG, elemS	chemical nature of mobile/reactive/gas/substrate particle
concM, concR, concG	fraction of the mobile/reactive/gas particles in the total flow
massM, massR, massG, massS	atomic mass of the mobile/reactive/gas/substrate particle [amu]
enMSw, enRSw, enGSw	energy switch for the mobile/reactive/gas particle: 0 ⇒ read energy distribution from a data file (e.g. eDist.txt) 1 ⇒ use the given energy mean value 2 ⇒ compute energy using an analytical function
enAvM, enAvR, enAvG	energy mean value for the mobile/reactive/gas particle [eV]
enFileM, enFileR, enFileG	energy distribution filename for the mobile/reactive/gas particle
angMSw, angRSw, angGSw	angle switch for the mobile/reactive/gas particle: 0 ⇒ read angular distribution from a data file (e.g. aDist.txt)

	1 ⇒ use the given values for θ_0, φ_0 and $\Delta\theta$ to compute the trajectory of an incoming particle 2 ⇒ uniform angular distribution
θ_0_M, φ_0_M, $\Delta\theta$_M, θ_0_R, φ_0_R, $\Delta\theta$_R, θ_0_G, φ_0_G, $\Delta\theta$_G,	angles defining the trajectory of an incoming mobile/reactive/gas particle [°]
angFileM, angFileR, angFileG	anglular distribution filename for the mobile/reactive/gas particle
T	deposition temperature [eV]: 75 °C = 0.03 eV
saveDataSw	save intermediate system structures switch: 0 – Not / 1 – Yes
saveDataNo	save intermediate system structures each "saveDataNo" deposition events
subBindEn	surface binding energy of a mobile particle with the substrate [eV]
layBindEn	surface binding energy of a mobile particle with the deposited structure [eV]
w_rot	substrate rotation speed [s^{-1}]
Ω_0	initial tilt angle of the substrate with respect to the z axis of the simulation box coordination system [°]
A	substrate oscillation amplitude [°]
w_osc	substrate oscillation frequency [s^{-1}]
maxDiffNo	forced deposition parameter: maximum number of diffusion events that can happen between two successive deposition events
*	*here, $\boldsymbol{nn_i}$ and $\boldsymbol{nn_f}$ stand for lateral nearest neighbour numbers of the initial and final position of a diffusing particle - more details are given below*

Innovative Coating Solutions,
Place Saint-Pierre, 11, B-5380 Forville., Belgium
slu@incosol4u.com www.incosol4u.com

13 Annex 2: Coordinate systems

There are two coordination systems used by NASCAM. First one is Cartesian coordinate system, where the coordinates of particles are given in real numbers (x, y, z). All the files with the information about the structure of the film (*.xyz files) use this system of coordinates. As these files are used for presenting the results of the simulations, we refer (x, y, z) system as an external coordination system. This coordination system is used to calculate the trajectory of the deposited particles also.

In the second coordinate system the positions of the atoms are given in integers numbers (i, j, k). All computations of dynamics of the system are done using this coordination system. For this reason we refer (i, j, k) system as an internal coordination system. The computations for structural analysis (island statistics, roughness…) are done using the internal coordination system too.

As it was already mentioned the particle coordinates are given in lattice parameter units.

The equations used to transform coordinates of the particles from one coordination system to the other one are given above in the Substrate section of this Manual. We remind to the reader that they are depending on the system (substrate) symmetry. For a given particle, considering its internal coordinates i, j and k, the corresponding external coordinates x, y, z are given by:

$$x = i - \frac{NXmax}{2}, \quad y = j - \frac{NYmax}{2}, \quad z = k,$$

for cubic symmetry and

$$x = i - \frac{NXmax}{2} + \frac{mod(j,2)}{2} \quad , \quad y = \frac{\left(j - \frac{NYmax}{2}\right)\sqrt{3}}{2}, \quad z = k,$$

for hexagonal symmetry. In the last case *mod(j,2)* equals to *0* if *j* is even and *mod(j,2)* equals to *1* if *j* is odd. For both symmetries *i* varies in between *0* and *NXmax-1* and *j* in between *0* and *NYmax-1*.

Figure A2.1 shows a graphical representation of the relative position of the internal (in blue) and external (in red) coordination systems, the polar and azimuthal deposition angles (θ and φ, respectively) defining the trajectory for an incoming particle (in green). Supposing that there is a pattern.txt file containing a single mask definition as "$\boldsymbol{i_m}$ $\boldsymbol{j_m}$ $\boldsymbol{di_m}$ $\boldsymbol{dj_m}$", the corresponding pattern surface is shown as well (in light grey bordered with blue).

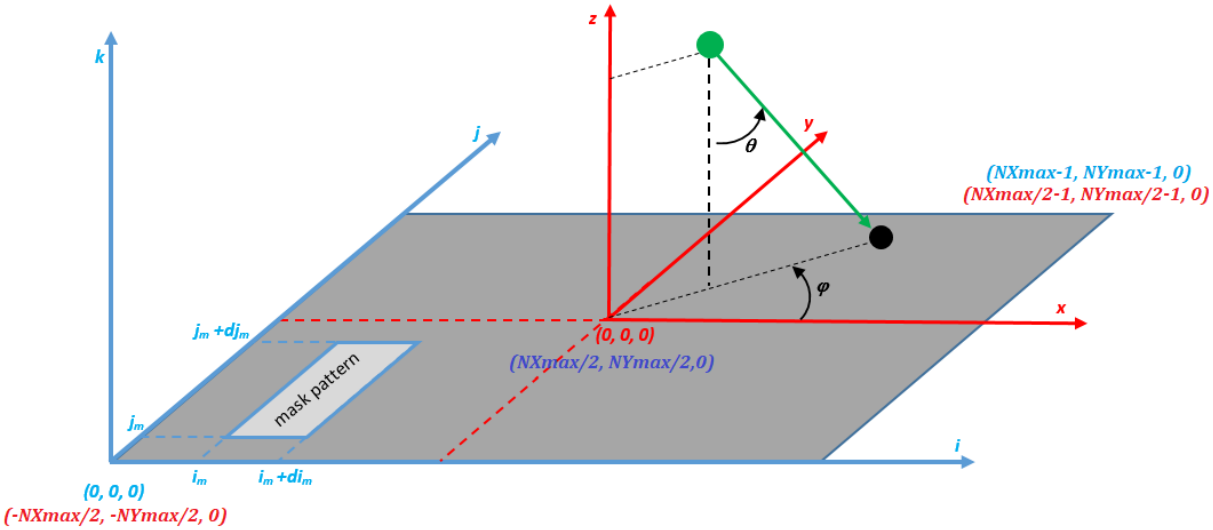


Figure A2.1. The relative position of the two coordination systems: internal and external one (in blue and in red, respectively). Some supplementary comments regarding the coordination systems are necessary:

- their origins are in the same horizontal plane ($k = 0$ and $z = 0$) that, in some situations, coincides with the initial substrate surface (in dark grey).
- the angles defining the trajectory of an incoming particle (in green) to its landing position (in black) are counted in respect with the external coordination system axes. In the figure, the final deposition angles θ and φ are shown. These angles are calculated according to the parameters that determine the trajectory of the incoming particles (given in the appropriate line of the *input.txt* file) and take into account all possible substrate movement, rotation, oscillation, as well as its possible initial tilting.
- a mask pattern (in light grey) corresponding to a "i_m j_m di_m dj_m" mask definition is shown as well. Its coordinates (and borders) are in blue to emphasize that the mask definition is considered in respect with the internal coordination system.