A Molecular Dynamics Study of the Energy of Activation of a Vacancy in an Ideal Xenon Monolayer and a Monolayer of Xenon Adsorbed on Ag(111)

by

Baghdad Barka

Thesis submited to the
School of Graduate Studies
in partial fulfillment of the requirements
for the degree of
Master of Science in Physics

Department of Physics
Faculty of Science
University of Ottawa
Ottawa, Ontario

ISBN 0-315-82583-9

To my family

ABSTRACT

We present a molecular dynamics study of the activation energy of a vacancy in a monolayer of xenon with two and three degrees of freedom, in the latter case in the presence of a Ag(111) substrate.

The large activation energy decreases with the third degree of freedom. The minimum energy trajectory between two lattice positions is given. In the three dimensional case the activated atom dips while the neighboring atoms rise.

With temperature the activation energy increases by about 10 percent from 5 to 65K in both the two dimensional and three dimensional cases.

The interaction between two vacancies has been investigated. This interaction is attractive, long-range and strong, similar in depth to the interparticle interaction.

ACKNOWLEDGEMENTS

My thanks are extended to the individuals who made this work possible.

My deepest thanks go to my supervisor, Professor Béla Joós for his helpful guidance, advice, support, understanding and encouragement during the preparation of this thesis .

I also wish to acknowledge the support and understanding of Professor G. Lamarche.

I thank also Bruno Grossman for his help to understand the main part of his version of the program of Molecular Dynamics Simulations.

In addition I should like to thank my wife for her help in English and her support, my family and special friends for their incessant encouragement and their great support.

The author is thankful to the government of Algeria for providing financial assistance in carrying out part of this research.

Contents

List of Figures

List of Tables

List of Symbols

Ag	The element Silver
$\mathcal{A}$	Area of the membrane
$\mathcal{A}_p$	Area of the perfect cell
$\mathcal{A}_d$	Area of the cell when the Vacancy is present
$\mathcal{A}_{Act}(i)$	Area of cell when the atom is activated
a	Lattice parameter
C_3, C_5	Parameters related to the potential $\mathcal{V}_{pol}$
C_6, C_8, C_{10}	Dispersion coefficients $\mathcal{V}(r)$
DR	Displacement of the activated atom in two dimensions
DX	Displacement of the activated atom along X axis
DY	Displacenent of the activated atom along Y axis
E	Energy
E_p	Potential energy for perfect cell
E_d	Potential energy when the Vacancy is present
E_v^f	Vacancy formation energy
$E_{Act}^t(i)$	The Activation Energy for each transition
$E_{Act}(i)$	The Total Activation Energy for the transition
$E_A(i)$	Energy when the atom at site P_2 is activated
E_{vac}	Total vacancy energy
E_s	Saddle point energy or maximum activation energy
$E_{Int}(r_s)$	Interaction energy between two vacancies in 2D
$E_{2vac}(r_s)$	Energy of 2 vacancies for different separations
$\{\mathbf{F}_k\}$	The force on the atom k
$\{\mathbf{F}_{kl}\}$	The force on atom k due to atom l
$\mathcal{H}(\mathbf{p}, \mathbf{q})$	The Hamiltonian
K	The area modulus

k_B	Boltzman Coefficient
$\mathcal{K}(\mathbf{p}, \dot{\mathbf{q}})$	Kinetic Energy
L_z	Height of atom above the surface of substrate
$\mathcal{L}(\mathbf{p}, \dot{\mathbf{q}})$	Lagrangian
m	Mass of atom for xenon
N	Number of atoms
n_{vac}	Number of vacancies
P	Calculated pressure for the system
P_E	External pressure
$\{\mathbf{p}_k\}$	Momenta
$\{\mathbf{p}'_k\}$	Projected Momenta
$\{\mathbf{q}_k\}$	The generalized coordinates
$\{\dot{\mathbf{q}}_k\}$	Time derivation of the generalized coordinates
$\mathcal{R}_i$	Random number
R_m	Length parameter at equilibrium
$\{\mathbf{r}_k\}$	Vectors position
$\{\mathbf{r}_{kl}\}$	$\mathbf{r}_{kl} = \mathbf{r}_k - \mathbf{r}_l$
T	Temperature
T_I	Instantaneous temperature
t	Time of simulation
U	Total potential
U_A	Defect energy at constant area
U_P	Defect energy at constant pressure
U_D	Total defect internal energy
V	Volume
$\{\mathbf{v}_k\}$	Vectors velocities
$\{\mathbf{v}'_k\}$	Projected velocities
$\mathcal{V}_{LJ}$	Lennard-Jones potential

$\mathcal{V}_{pol}$	Potential due to polarisation
$\mathcal{V}_s$	Potential given by the contribution of s-r repulsion
Xe	Rare gas xenon
z_0	Constant related to adatom-surface separation
$\{\mathbf{z}_k\}$	Position of atoms along Z axis
$\langle Z \rangle$	Average of Z coordinate of the atom
β	Scaling factor
β_{xy}	Scaling factor in the XY plan
β_z	Scaling factor in Z direction
ε	represents an energy
$\{\rho_\mathbf{k}, \dot{\rho}_\mathbf{k}\}$	Scaled variables
σ	Effective radius of the potential
Δ	Distance related to to multilayer physisorption
Δa	Variation of lattice parameter
Δt	Molecular dynamics step
ΔU	Change in energy of the total system
Δz	Radius of the potential wall
∇_i	Gradient with respect to molecular positions
θ_D	Debye temperature
2D	Two dimensions
3D	Three dimensions.

Chapter 1

INTRODUCTION:

The recent progress in Condensed-matter physics has permitted the development of materials, experimental and theoretical techniques for studying monolayer films on uniform surfaces. This has attracted the interest of many theorists and experimenters throughout the world.

The first systems to show clear two-dimensional gas-like features were semi-soluble oil-on-water films, in studies that [1] began in the 19th Century and continued after that.

The simplicity of the rare gases has allowed straight-forward intuitive considerations as well as explicit microscopic calculations. Many techniques have been applied to rare gas monolayers and bilayers so that a lot of detailed information about the structure and the phases is available. The two-dimensional rare gas solids have continued to be an active and productive area of research. There have been many studies of adsorbed rare gases on substrates particularly, graphite. The introduction of dynamical simulations [2,6,7] helped the discovery of many interesting properties of these systems.

Important questions remain unanswered after all these experiments.

In the study of atomic monolayer and in particulary rare-gases, one of the important points is the presence of vacancies which are common in monolayers, more so than in bulk solids because thermodynamic equilibrium in the former involves exchange with the external environment as well as entropic considerations. Vacancies are particularly abundant in physisorbed monolayers, in which the concentration can be as high as 10 percent before the melting point is reached [3].

The perception was that because it is a local defect its influence should be small. The role of these vacancies until now has been ignored. But recently it was shown that the presence of vacancies at concentrations of just a few percent can have profond consequences on the structural properties of a monolayer below the melting point [4]. The interest of this, is that these vacancies could play a part in phase transitions.

The important question which we ask is: **With what facility does the vacancy move on the surface of the monolayer with and without a substrate ?** Which is important in diffusion phenomena. A key parameter in determining the diffusion rate is the "Vacancy migration energy" or "Activation energy of a vacancy". This is the subject of my research.

The vacancy migration energy was introduced by Johnson [5]. In his work he simulates the positional interchange between a vacancy and one of its nearest-neighbor atoms. He also led the discussion on migration energy calculation techniques.

We used the same procedure to study the minimum energy trajectory of an atom between two sites for the two cases:

First for a two dimensional ideal xenon monolayer without a substrate. Secondly for a three dimensional xenon monolayer adsorbed on the surface (111) of silver.

As well we describe the temperature dependence of these energies.

The calculations of the variables of interest were made by a molecular dynamics simulation program.

The procedure to calculate this activation energy consist first in creating a vacancy in our system and letting the system reach a thermal equilibrium in presence of this defect. The second step is to move one of the nearest-neighbor atoms towards the vacancy and obtaining an equilibrium state for different displacements of the atom. This generates the minimum energy trajectory of an atom from one site to the other. The maximum in this trajectory is the activation energy of the vacancy. The details of this calculations as well as for the vacancy energy are given in chapter 5.

The monolayers can appear in many forms with various interactions. We focus on monolayers formed of atoms interacting with spherical potentials of Lennard-Jones form, specifiquely Aziz form, see chapter 2 for details.

The last part of our work considers the effects of the modification of the interaction between a defect and its neighboring atoms due to the presence of another defect, in our case the defect represent a vacancy. This was presented by the interaction between two vacancies. An introduction and a historical perspective of this problem is given in the introduction of chapter 7.

This interaction between two vacancies was described by the difference between the energy of two vacancies and two times the energy of one vacancy. This problem was treated only for a two dimensional xenon monolayer.

Chapter 2 : Provides some knowledge of our system of interest and the interaction potentials used to describe the interaction between xenon atoms and the interaction of each xenon atom with the surface of the substrate which is silver.

Chapter 3 : Presents a summary of the dynamics simulation methods for different microcanonical and canonical ensembles.

Chapter 4 : Describes the model and the conditions used for our calculations. This chapter also provides the conditions for choosing the method of simulation.

Chapter 5 : Describe the energy calculation methods for the different energies (Vacancy energy, Activation energy of a vacancy) and gives an introduction of the definition of the activation energy.

Chapter 6 : Result and discussion of the activation energy, vacancy energy and the temperature dependence of these energies.

Chapter 7 : Presents an introducion and the definition to the interaction between two vacancies in a two dimensional Xe monolayer as well as the results and a discussion.

Finally, we give a summary of the conclusions that can be drawn from our calculations.

Chapter 2

SYSTEM OF INTEREST AND INTERACTION POTENTIALS

2.1 Rare Gases :

Between 1894 and 1898 Lord Rayleigh, W. Ramsay and W. M. Travers [8] discovered the elements helium (He), neon (Ne), argon (Ar), krypton (Kr) and xenon (Xe) known as **RARE GASES**.

The condensed rare gases have been extensively studied for the last half century. The first application in the concept of binding, crystal structure and lattice dynamics in most elementary texbooks of physics and chemistry are made using condensed argon, krypton and xenon. This is because the interactions are simple and well known. Only interactions between pairs of atoms need to be considered. The interactions are spherically symmetric with a Van der Waals interaction .

2.2 Adsorbed Rare Gases:

2.2.1 Introduction:

The adsorbed gas systems depart from ideal two-dimensional systems in two respects : They have important interactions with the substrates, and they display phenomena associated with the formation of more than one layer of adsorbate.

The adsorbed gases provide an almost inexhaustible number of possible combinations of substrate and adsorbate. There have been many studies of such systems.

2.2.2 Experiments:

In the late 1960's Thomy and Duval carried out accurate vapor pressure isotherm measurements of krypton and xenon physisorbed onto the basal planes of graphite. These measurements revealed that the rare gas atoms adsorb onto the surface, atomic layer by atomic layer and the density on the surface may be controlled in equilibrium .

Much of the recent experiments have dealt with adsorption on graphite. As a result they found that in the graphite system the modulation of the adsorbate atom potential by the discrete structure of the surface has important consequences and leads to registry-disregistry transitions and rotational alignment. These effects have been observed and have been treated theoretically by

[9,10,11,12,13,14].

When noble gases are adsorbed on a metal surface, the ordered phases may be either in or out of registry with the substrate structure depending on the metal and its surface orientation.

Some of the experiments were done for the Xe incommmensurate triangular lattice on the close packed (111) surface of silver . This interesting system has shown that there are no effects of the lateral forces on the adsorbed atom from the discrete substrate structure. It appears that the gas is adsorbed on a smooth dielectric continuum. This then provides a simpler test of our understanding of adatom-adatom interactions and the statistical mechanics of the 2D phases without the complications of the registry forces.

Chesters [15] first observed that Xe monolayer on Ag was a triangular lattice incommensurate but aligned with the substrate.

Cohen, Unguris and Webb [16] studied this system with low energy electron diffraction. They observed that the lattice compressed as the coverage approached one.

Recent synchrotron studies of xenon on the (111) face of silver [17] provide further evidence that xenon melts into an intrinsic hexatic phase.

Several questions remain after all these experiments which indicated that further experiments on the Xe-Ag(111) surface are worthwhile.

2.3 Potentials of interaction :

We can represent the total potential energy U for a given configuration of N Xenon atoms $\mathbf{r}_i$, (i=1,2,.......,N) above a substrate surface of Silver in the form [18] :

$$U = \sum_{i>j}^{N} \Psi_{Xe-Xe}(|\mathbf{r}_i - \mathbf{r}_j|) + \sum_{i=1}^{N} \Psi_{Xe-Ag(111)}(z_i) \quad (2.1)$$

Where the surface (111) of silver is represented in the plan (X, Y) and z_i is the positions of the xenon atoms along Z axis from the surface of the substrate. Simple pair-wise additivity of the interatomic interactions is assumed for the first term xenon-xenon interaction. In order to reduce computational time in the evaluation of the potential energy, certain approximations and procedures were implemented.

The second term of the equation (2.1) represents the interaction of each xenon atom with the surface (111) of silver. We choose surface (111) of silver as substrate because we know that the potential of absorption for silver is almost flat, i.e. the potential of interaction of xenon atoms with the surface of silver can be given as a function only of the separation of each atom of xenon with the surface of the substrate.

The Hamiltonian describing an assembly of particles adsorbed on a surface has two ingredients: the mutual interaction between adsorbed molecules of Xe which represents the first term of the equ. (2.1) and the interaction of each particle with the surface of the substrate in our case Ag .

2.3.1 Adatom-adatom Potentials:

As first approximation, we could use the **Lennard-Jones** 6-12 pair potential:

$$\mathcal{V}_{LJ}(r) = 4\,\varepsilon\,\left[\left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^{6}\right] \tag{2.2}$$

where,

ε = represents an energy,

σ = represents the effective radium of the potential,

$(\frac{\sigma}{r})^{12} \Rightarrow$ repulsive term,

$(\frac{\sigma}{r})^{6}$ = dipole-attraction, most important part of the potential

The parameters ε and σ for the potential (2.2) are given for the interaction Xe-Xe in table 2.1 [19]

We can obtain a potential of greater precision for our system Xe-Xe. The bare pair potentials have been chosen to follow the Aziz form [20], which consists of a short-range closed-shell repulsion derived from a self-consistant Hartree-Fock calculations (Virial Coeficients), the potential will be represented by the

Table 2.1: Parameters for the Morse-Van der Waals model of the rare-gas Xe-Xe interaction

Parameters	$\varepsilon(K)$	$\sigma(\mathring{A})$
Xe	214	3.92

expression:

$$\mathcal{V}(r) = \varepsilon \left[A\, x^{\gamma} \exp(-\alpha x) - \left(\frac{C_6}{x^6} + \frac{C_8}{x^8} + \frac{C_{10}}{x^{10}}\right) F(x)\right] \tag{2.3}$$

where,

$$F(x) = \begin{cases} \exp[-(\frac{D}{x} - 1)^2] & \text{if } x < D \\ 1 & \text{otherwise} \end{cases} \tag{2.4}$$

and,

$$x = \frac{r}{R_m} \tag{2.5}$$

R_m is the separation for the minimum in the two particle interaction potential. C_6,C_8 and C_{10} the dispersion coefficients. In the case of the Xe-Xe inter-

Table 2.2: Parameters for the interatomic potentials

Parameters	Xe-Xe
ε/k_B(K)	282
$R_m(\mathring{A})$	4.36
A	0.1215312 10^8
α	16.496763
γ	2.40
C_6	1.1561739
C_8	0.5414923
C_{10}	0.2839735
D	1.28

action we used the Kr-Kr potential [20] scaled to an appropriate well depth $\varepsilon/k_B = 282K$ and length parameter $R_m = 4.36\mathring{A}$ as in [21].

F and x^γ explain the contribution of short-range and long-range interactions. The resulting potential parameters are given in table (2.2) [22,23,24].

The potential used for the interaction Xe-Xe is given by figure (2.1).

The effective range of the full pair potential is 3a in our case, about 13$\mathring{A}$ [25], which includes 36 neighbours in the perfect lattice. This is one of the approximations and procedures used to reduce computational time.

In the two-dimensional ideal monolayer, the potential U which is given by the equation (2.1) will be represented by $\mathcal{V}(r)$.

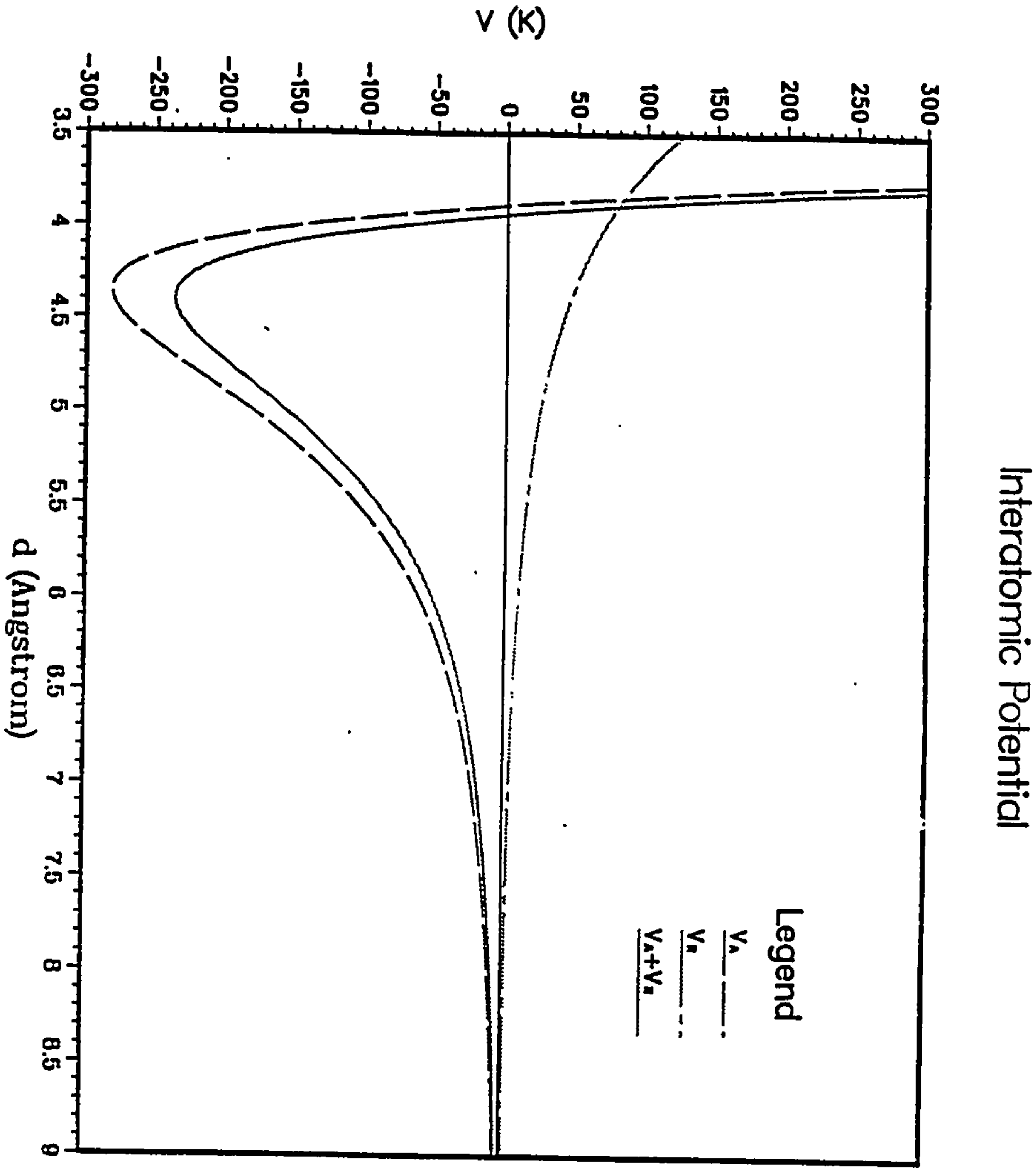

Figure 2.1: Xenon pair potentials, consist of two parts the repulsive part (dash-dot curve) and the attractive part (dash curve). The solid curve used for calculations represents the total potential.

2.3.2 Surface Adatom Potentials:

The interaction of a physisorbed atom with a surface is represented by the potential $\mathcal{V}_z$ which is given by the contribution of short-range repulsion, $\mathcal{V}_s$, and long-range attraction, $\mathcal{V}_{pol}$.

The potential $\mathcal{V}_{pol}$ is due to polarisation and is analogued to the Van der Waals dispersion interaction:

$$\mathcal{V}_{pol}(r) \longrightarrow -C_3/z^3 \tag{2.6}$$

this potential is independant of the microscopic surface structure , C_3 depends upon the macroscopic properties of the surface and the characteristics of the adatom. In case of bilayers and trilayers which is not our case, the potential $\mathcal{V}_{pol}$ can be represented by:

$$\mathcal{V}_{pol} = -\frac{C_3}{(z-\Delta)^3} - \frac{C_5}{(z-\Delta)^5} \tag{2.7}$$

In this case the origin of the $1/z^3$ and $1/z^5$ behaviour is the distance Δ, where $\Delta \approx d(111)/2 \simeq 1.18\mathring{A}$, the parameters C_3 and C_5 will be given in table(2.3) for the system Xe-Ag(111) [26,27].

The procedure assumes that $\mathcal{V}_s$ which represents the short-range potential is proportional to the surface charge density [28] and suggests that $\mathcal{V}_s$ has the form:

$$\mathcal{V}_s = A\ \exp(-\eta\,(z-z_0) - D(x,y)) \tag{2.8}$$

where $D(x,y)$ contains the surface corrugation effects , in the case of Xe physisorbed on the compact Ag(111) surface we ignore this factor.

We choose to represent $\mathcal{V}_s + \mathcal{V}_{pol}$ by the Morse function:

$$\mathcal{V}_m = \varepsilon\left[\exp(-\eta\,(z-z_0)) - 2\ \exp(-\eta\,(z-z_0)/2)\right] \tag{2.9}$$

$\mathcal{V}_m$ will be used in monolayer adatoms, while bilayers and trilayers are assumed to interact with the surface via $\mathcal{V}_{pol}$ given by the equation (2.7).

The parameters of $\mathcal{V}_m$ are given in table(2.3). We note that ε is closely related to the latent heat of absorption, z_0 is the minimum in the adatom-surface separation and η is determined by the Einstein-like photon frequency for the monolayer [29]. $\mathcal{V}_m$ is shown in figure(2.2).

Adatom-surface potentials for the monolayers are represented by a Morse function whose well depth and curvature are parametrized to produce the latent heat of condensation and the frequency of the dispersionless phonon branch with atomic displacements perpendicular to the surface. The parameters of $\mathcal{V}_m$ have been obtained in a self-consistent fashion using the measured properties of the monolayer.

In the three-dimensional adsorbed monolayer, the potential U will be represented by $\mathcal{V}(r) + \mathcal{V}_m(r)$.

Table 2.3: Parameters for the Morse-Van-der Waals model of the rare-gas Xe-Ag(111) interaction (1MeV = 11.6 k_B(K)).

Parameters	Xe-Ag(111)
ε(MeV)	159.6
η	0.92
z_0	3.53
C_3	3590
C_5	6390
Δ	1.20

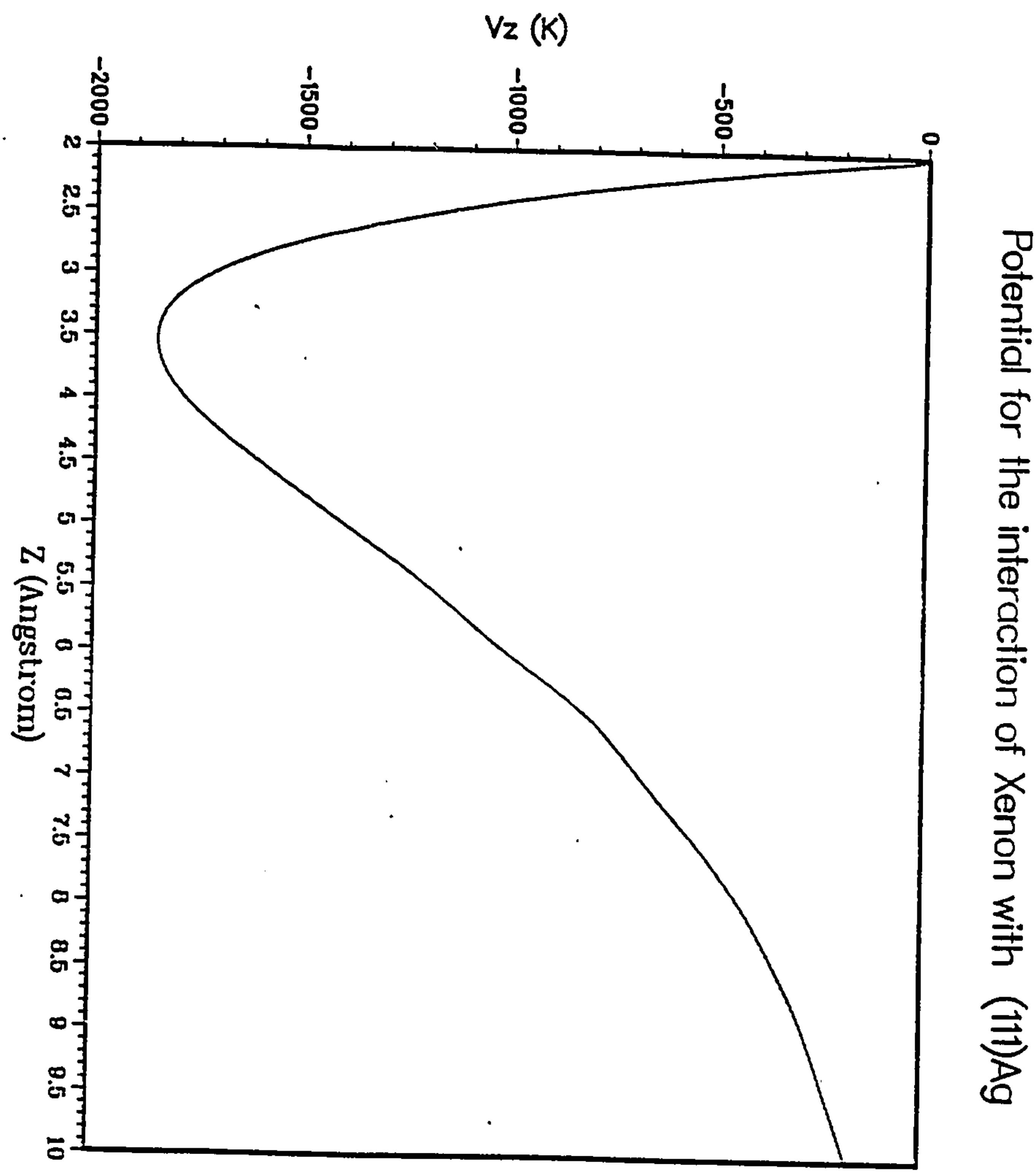

Figure 2.2: Morse-Van der Waals potential for the interaction of Xenon with the (111) surface of silver.

Chapter 3

MOLECULAR DYNAMICS SIMULATIONS METHODS

3.1 Introduction :

In recent years it has become possible to solve Newton's equation of motion for systems of several interacting particles and to investigate the dynamical correlations in complete microscopic detail of the phase trajectory of the system. This technique is usually referred to as **Molecular Dynamics.** The first molecular dynamics calculations were made by Alder and Wainwright [30] using hard-sphere and square-well potentials.

We will treat some interesting methods which relate to the dynamics of atomic systems.

3.2 Equations of motion for atomic systems:

In this chapter we try to define in brief the techniques of molecular dynamics used to solve the classical equations of motion known as Newton equations for a system of N atoms interacting via a potential U as in equation (2.1);
The fundamental form is:

$$\frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot{\mathbf{q}}_k}\right) - \left(\frac{\partial \mathcal{L}}{\partial \mathbf{q}_k}\right) = 0 \tag{3.1}$$

Known as the Lagrangian equation of motion.
The Lagrangian $\mathcal{L}(\mathbf{q}, \dot{\mathbf{q}})$ is defined as :

$$\mathcal{L}(\mathbf{q}, \dot{\mathbf{q}}) = \mathcal{K}(\mathbf{q}, \dot{\mathbf{q}}) - U(\mathbf{q}) \tag{3.2}$$

$$\mathcal{K}(\mathbf{q}, \dot{\mathbf{q}}) = \sum_{k=1}^{N} \frac{\mathbf{p}_k^2}{2\, m_k} \tag{3.3}$$

Where $\mathcal{K}(\mathbf{q}, \dot{\mathbf{q}})$ is the Kinetic energy and $U(\mathbf{q})$ is the potential energy given by the equation (2.1). $\mathbf{q}_k$ are the generalized coordinates, $\dot{\mathbf{q}}_k$ their time derivatives and $\mathbf{p}_k$ the generalized momentum conjugate to $\mathbf{q}_k$. The momenta are defined as :

$$\mathbf{p}_k = \frac{\partial \mathcal{L}}{\partial \dot{\mathbf{q}}_k} \tag{3.4}$$

If we consider the Cartesian coordinates $\mathbf{r}_k$ the equation (3.1) becomes:

$$\mathbf{F}_k = m_k\, \ddot{\mathbf{r}}_k \tag{3.5}$$

Where m_k and $\mathbf{F}_k$ the mass and the force of the atom k. The force of the atom is defined as:

$$\mathbf{F}_k = \frac{\partial \mathcal{L}}{\partial \mathbf{r}_k} = -\frac{\partial U}{\partial \mathbf{r}_k} \tag{3.6}$$

The Hamiltonian in the coordinates $(\mathbf{p}, \mathbf{q})$ is given by:

$$\mathcal{H}(\mathbf{p}, \mathbf{q}) = \sum_{k=1}^{N} \dot{\mathbf{q}}_k\, \mathbf{p}_k - \mathcal{L}(\mathbf{q}, \dot{\mathbf{q}}) \tag{3.7}$$

Finally the Hamiltonian equations of motion are :

$$\frac{d\mathbf{r}_k}{dt} = \frac{\partial \mathcal{H}}{\partial \mathbf{p}_k} = \frac{\mathbf{p}_k}{m} \tag{3.8}$$

$$\frac{d\mathbf{p}_k}{dt} = -\frac{\partial \mathcal{H}}{\partial \mathbf{r}_k} = -\frac{\partial U(\mathbf{r}_k)}{\partial \mathbf{r}_k} \tag{3.9}$$

All the algorithms start with the same equations of motion, Newton's equations.

3.3 Molecular Dynamics methods:

The molecular dynamics methods operate most naturally in the microcanonical ensemble (Rahman [31], Verlet [32]). Extensions of these methods to other ensembles have been introduced in recent years (Kushik and Berne [33], Andersen [34]). The particle motions in molecular dynamics simulations are determined by numerically integrating the equations of motion . The time increments in the integration are called time steps and involve moving each particle. The choice of the length of these steps must be made so that they are in the order of real microscopic time scales of the system. In our work we ran the program for 5000-10000 integrations steps of length 2.0 10^{-14} seconds in two dimensions, in three dimensions we used a length of 1.0 10^{-14} seconds for more detail.

3.3.1 Molecular Dynamics Algorithm:

All the techniques have the same procedure and can be described by the same algorithm as followed :

Algorithm:

Step 1	:	Specify the initial positions $\mathbf{r}_k(t)$ and the initial velocities $\mathbf{v}_k(t)$ [SETIS, POTEM, DISVIT].
Step 2	:	Evaluate the force from the potential energy and hence acceleration $a_i = f_i/m_i$, from the current positions [FORCE, FZ].
Step 3	:	Obtain new positions, velocities, accelerations etc.,at time $t + \Delta t$, using the current values of these quantities. [MOUVE2 or MOUVE3].
Step 4	:	Calculate any variables of interest, such as the Energy, Pressure, Lattice-parameter, etc. Prepare the accumulation of time averages, before returning for the next step Δt [FORCE, FZ , MOUVE2]. Return to "Step 2".

[SETIS, POTEM, DISVIT, FORCE, FZ, MOUVE2, etc.] see Appendix B.

3.3.2 Microcanonical (N,V,E) ensemble :(Verlet)

Ensemble of constant number of particles, N, constant volume, V, and constant energy, E, the equations of motion were integrated using the "leapfrog" form of the Verlet algorithm [32].

$$\mathbf{v}_k(t+\frac{\Delta t}{2}) = \mathbf{v}_k(t-\frac{\Delta t}{2}) + \mathbf{F}_k(t)\,\frac{\Delta t}{m}, \tag{3.10}$$

$$\mathbf{r}_k(t+\Delta t) = \mathbf{r}_k(t) + +\mathbf{v}_k(t+\frac{\Delta t}{2})\,\Delta t, \tag{3.11}$$

$$\mathbf{v}_k(t) = \frac{\mathbf{v}_k(t-\frac{\Delta t}{2}) + \mathbf{v}_k(t+\frac{\Delta t}{2})}{2}, \tag{3.12}$$

Where,

$$\mathbf{v}_k(t-\frac{\Delta t}{2}) = \frac{\mathbf{r}_k(t) - \mathbf{r}_k(t-\Delta t)}{\Delta t} \tag{3.13}$$

The modifications to the basic Verlet algorithm which is called half-step "leapfrog" have been proposed for programming advantages. Numerical benefits derive from the fact that at no stage do we take the difference of two large quantities to obtain a small one; this minimizes loss of precision on a computer.

If we introduce (3.10) and (3.13) into (3.11) it becomes;

$$\mathbf{r}_k(t+\Delta t) = 2\,\mathbf{r}_k(t) - \mathbf{r}_k(t-\Delta t) + \mathbf{F}_k(t)\,\frac{\Delta t^2}{m} \tag{3.14}$$

With this algorithm the velocities are not required but they may be evaluated by the central difference formula.
We keep repeating this procedure for NPAS steps until we reach the equilibrium. The energy E is the total energy (potential and kinetic).

3.3.3 Canonical (N,V,T) ensemble :(Rescaling of velocities)

The only modification is that velocities at the previous half-time step are scaled;

$$\mathbf{v}_k(t + \frac{\Delta t}{2}) = \mathbf{v}_k(t - \frac{\Delta t}{2})\,\beta + \mathbf{F}_k(t)\,\frac{\Delta t}{m}, \tag{3.15}$$

The equilibrium (thermal equilibrium) process is performed by integrating the equation of motion for a certain number of MD steps, the integration process is then stopped and energy removed or added by a readjustment of the velocities to maintain constant temperature, and the process continues. This is done by scaling all the velocities with the parameter β;

$$\beta^2 = \frac{3\,(N-1)\,k_B\,T/m}{\sum_{k=1}^{N} \mathbf{v}_k^2(t - \frac{\Delta t}{2})} \tag{3.16}$$

for the positions $\mathbf{r}_k(t + \Delta t)$ same as equation (3.11).

3.3.4 Canonical (N,V,T) ensemble :(Damped Force Method)

The Hamiltonian equations will be modified;

$$\frac{d\mathbf{p}_k}{dt} = \mathbf{F}_k - \alpha\,\frac{\mathbf{p}_k}{m}, \tag{3.17}$$

$$\frac{d\mathbf{q}_k}{dt} = \frac{\mathbf{p}_k}{m} \tag{3.18}$$

α is a constant , in this case we require for a constant temperature that;

$$\frac{dT}{dt} = 0 \tag{3.19}$$

since

$$T = \frac{\sum_{k=1}^{N} \mathbf{p}_k\,\mathbf{p}_k}{3\,m\,(N-1)\,k_B}, \tag{3.20}$$

the derivative of T gives;

$$\frac{dT}{dt} = \frac{2\,\sum_{k=1}^{N} \frac{d\mathbf{p}_k}{dt}\,\mathbf{p}_k}{3\,m\,(N-1)\,k_B}, \tag{3.21}$$

if we substitute (3.17) into (3.21);

$$\frac{dT}{dt} = \frac{2\,\sum_{k=1}^{N}(\mathbf{F}_k\,\mathbf{p}_k - \alpha\,\frac{\mathbf{p}_k}{m}\,\mathbf{p}_k)}{3\,m\,(N-1)\,k_B}, \tag{3.22}$$

and from the equation (3.19) and (3.22) we get;

$$\alpha = \frac{m\,\sum_{k=1}^{N}\mathbf{F}_k\,\mathbf{p}_k}{\sum_{k=1}^{N}\mathbf{p}_k\,\mathbf{p}_k} \tag{3.23}$$

The equation (3.10) now becomes;

$$\mathbf{v}_k(t+\frac{\Delta t}{2}) = \mathbf{v}_k(t-\frac{\Delta t}{2}) + \mathbf{F}_k(t)\,\frac{\Delta t}{m} - \alpha\,\mathbf{v}_k(t)\,\frac{\Delta t}{m} \tag{3.24}$$

the projected velocity $\mathbf{v}'_{\ k}$ which is different from the constrained velocity $\mathbf{v}_k$ is given by;

$$\mathbf{v}'_{\ k}(t) = \mathbf{v}_k(t-\frac{\Delta t}{2}) + \mathbf{F}_k(t)\,\frac{\Delta t}{2\,m} \tag{3.25}$$

if we substitute the equations (3.12),(3.15),(3.24) and (3.25) we can find a relation between α and β;

$$\beta = \frac{1}{1+\alpha\,\frac{\Delta t}{2\,m}} \tag{3.26}$$

the equation (3.24) becomes;

$$\mathbf{v}_k(t+\frac{\Delta t}{2}) = \mathbf{v}_k(t-\frac{\Delta t}{2})\,(2\,\beta-1) + \beta\,\mathbf{F}_k(t)\,\frac{\Delta t}{m} \tag{3.27}$$

in this situation we do not need to calculate α. The parameter β for the equation (3.27)is given by;

$$\beta^2 = \frac{3\,(N-1)\,k_B\,T_c/m}{\sum_{k=1}^{N}\ \mathbf{v}'^{\,2}_{\ k}(t)} \tag{3.28}$$

This method ensures that T(t) is constant at every time step.

3.3.5 Canonical (N,P,$\mathcal{H}$) ensemble :(Andersen's Method)

Andersen's method [34] consists of maintaining the pressure for any given system.

The Hamiltonian for the system is given in terms of scaled variables as new coordinates:

$$\mathcal{H}(\{\rho_k\},\{\dot{\rho}_k\},V,\dot{V}) = \frac{V^{2/3}}{2}\sum_{k=1}^{N} m\dot{\rho}_k\dot{\rho}_k + \sum_{k=1}^{N}\sum_{k>l}^{N}\Phi(\rho_{kl}V^{1/3}) + \frac{1}{2}M\dot{V}^2(t) + P_E V(t) \tag{3.29}$$

Where V is the volume of the system, M is a constant which behaves as an inertial "mass" for the change in volume. P_E is the external pressure. The first two terms of the Hamiltonian (3.29) are equivalent to the internal energy of the particles of the system. The third term is a kinetic energy for the motion of V, and the fourth represents a potential energy associated with V.
A physical interpretation can be given to the Hamiltonian. If we suppose that example our system is a fluid in a container of variable V. The fluid can be compressed by a piston, $P_E V$ is the potential derived from an external pressure P_E acting on the piston, and M is the mass of the piston. Our Hamiltonian is a form of enthalpy. As it is known that the enthalpy is the appropriate thermodynamic potential that needs to be minimized for a process done at constant pressure.

The ensemble average $\langle\mathcal{H}(t)\rangle$ differs from the enthalpy of the N-particule system by $\frac{1}{2}\,k_B\,T$ which is the average kinetic energy associated with the volume fluctuations i.e $\langle\frac{1}{2}\,M\,\dot{V}^2(t)\rangle$.
The scaled variables (ρ_k, $\dot{\rho}_k$) can be expressed as:

$$\rho_k = \frac{\mathbf{r}_k}{V^{1/3}} \tag{3.30}$$

$$\frac{d\rho_k}{dt} = \frac{\mathbf{p}_k}{m\,V^{1/3}} \tag{3.31}$$

Each component of ρ_k is a dimensionless number between zero and one.

The equations of motion will now be coupled. In Newtonian formula they are:

$$\frac{d^2\rho_k}{dt^2} = \frac{\mathbf{F}_k}{m\,V^{1/3}} - \frac{2}{3}\frac{d\rho_k}{dt}\left(\frac{d\ln V}{dt}\right), \tag{3.32}$$

$$\frac{d^2V}{dt^2} = \frac{P - P_E}{M} \tag{3.33}$$

With P calculated from the virial theorem;

$$P = \frac{1}{3\,V}\left(\sum_{k=1}^{N}\frac{1}{m}\,\mathbf{p}_k\,\mathbf{p}_k + \sum_{k=1}^{N}\sum_{k>l}^{N}\mathbf{r}_{kl}\,\mathbf{F}_{kl}\right) \tag{3.34}$$

Where $\mathbf{r}_{kl} = \mathbf{r}_k - \mathbf{r}_l$ and F_{kl} is the force on atom k due to atom l.

P represents the calculated pressure in the system for each step of time. At equilibium the average $\langle P(t)\rangle$ is equal to P_E for constant pressure. (see Appendix A for calculation details.)

In equations (3.32) and (3.33) the volume plays the role of simply another dynamic variable, while the volume fluctuations do not disturb the equilibrium of the system.

The molecular dynamics procedure in this case will be described in two steps: first, is to solve the coupled equations of motion, Eqs. (3.32) and (3.33), the second step in the procedure is to transform the trajectory from the scaled coordinates to the unscaled coordinates using Eqs. (3.30) and (3.31).

We cannot integrate the equation (3.32) because of the term $\dot{\rho}_k$. However, this difficulty can be overcome by transforming the equation into cartesian coordinates.

$$\frac{d\rho_k}{dt} = \frac{1}{V^{1/3}}\left(\frac{d\mathbf{r}_k}{dt} - \frac{1}{3}\,\mathbf{r}_k\,\frac{d\ln V}{dt}\right) \tag{3.35}$$

from the equation (3.31) we get;

$$\frac{\mathbf{p}_k}{m} = \frac{d\mathbf{r}_k}{dt} - \frac{1}{3}\,\mathbf{r}_k\,\frac{d\ln V}{dt} \tag{3.36}$$

A similar problem exists in the integration of (3.33). The approximation used is applying Verlet's algorithm to momenta rather than positions and ignoring the term in Δt^2 . This gives us :

$$V(t+\Delta t) = 2\,V(t) - V(t-\Delta t) + (P(t) - P_E)\,\frac{\Delta t^2}{M}, \tag{3.37}$$

$$\frac{dV(t)}{dt} = \frac{(V(t+\Delta t) - V(t-\Delta t))}{2\,\Delta t}. \tag{3.38}$$

The "leapfrog" algorithm for updating the velocities becomes;

$$\mathbf{v}_k(t+\frac{\Delta t}{2}) = \mathbf{v}_k(t-\frac{\Delta t}{2}) + \left(\frac{\mathbf{F}_k(t)}{m} + \frac{\mathbf{r}_k(t)}{3\,V(t)}\left[\frac{d^2V(t)}{dt^2} - \frac{2}{3}\left(\frac{d\ln V(t)}{dt}\right)^2\right]\right)\Delta t. \tag{3.39}$$

The other equations stay the same.

The value of M affects the fluctuations in the unscaled volume V, this value will also effect the rate at which the system approaches equilibrium.
Andersen has suggested that it would be appropriate to choose M so that the time-scale for the volume fluctuations is approximately equal to the time taken for a sound wave to travel from one end of the sample to the other. His techniques will, paradoxically, always lead to fluctuations in pressure greater than those in a constant volume simulation.
The results of Haile and Graben [35] indicate that static properties are relatively insensitive to the change of the value of M . Choosing M to be too small can lead to very rapid fluctuations in the volume.

3.3.6 Canonical (N,P,T) ensemble:(Combined Method)

This method is the combination of the two last methods (N,P,$\mathcal{H}$) and (N,V,T). The projected velocities are calculated using

$$\mathbf{v}'_k(t) = \mathbf{v}_k(t - \frac{\Delta t}{2}) + \left(\frac{\mathbf{F}_k(t)}{m} + \frac{\mathbf{r}_k(t)}{3\,V(t)} \left[\frac{d^2V(t)}{dt^2} - \frac{2}{3} \left(\frac{d\ln V(t)}{dt} \right)^2 \right] \right) \frac{\Delta t}{2}. \tag{3.40}$$

The updating velocities from "leapfrog" algorithm becomes;

$$\begin{aligned} \mathbf{v}_k(t + \frac{\Delta t}{2}) &= \mathbf{v}_k(t - \frac{\Delta t}{2})\,(2\,\beta - 1) + \left(\frac{2}{3}\,\mathbf{r}_k(t)\,\frac{d\ln V(t)}{dt} \right)\,(1-\beta) \\ &+\beta \left(\frac{\mathbf{F}_k(t)}{m} + \frac{\mathbf{r}_k(t)}{3\,V(t)} \left[\frac{d^2V(t)}{dt^2} - \frac{2}{3} \left(\frac{d\ln V(t)}{dt} \right)^2 \right] \right) \Delta t. \end{aligned} \tag{3.41}$$

Where the scaling factor, β will be;

$$\beta^2 = \frac{3\,(N-1)\,k_B\,T_c/m}{\sum_{k=1}^{N}(\frac{\mathbf{p}'_k(t)}{m})^2} \tag{3.42}$$

The equation (3.36) is use to calculate the projected momenta at time t.

$$\frac{\mathbf{p}'_k}{m} = \mathbf{p}'_k(t) - \frac{1}{3}\,\mathbf{r}_k\,\frac{d\ln V}{dt} \tag{3.43}$$

In this method the velocity of a particle is dependent upon its positions.

$$\mathbf{v}_k = \frac{\mathbf{p}_k(t)}{m} - \frac{1}{3}\,\mathbf{r}_k\,\frac{d\ln V}{dt} \tag{3.44}$$

Therefore, both the position and velocity of a particle have to be altered if a boundary is crossed.

The value of the parameter M can be changed after a number of time steps by a value smaller than the first one to reduce the fluctuations of our quantities energy, lattice-parameter and energy as shown in figures 4.27 to 4.32 .

Chapter 4

MODEL AND METHOD OF SIMULATION CHOSEN

4.1 Introduction:

The system of interest to be simulated is rare-gas monolayer. The system contain N atoms, with coordinates $\mathbf{r}_1, \mathbf{r}_2, ..., \mathbf{r}_N$ in a triangular lattice as showed in figure 4.1 above a substrate surface (111) of silver in the three dimensional case which is represented in (X, Y) plan, with periodic boundary conditions. The system has been treated at constant temperature where the energy fluctuates.

The constant pressure is obtained by iteration with respect to area change.

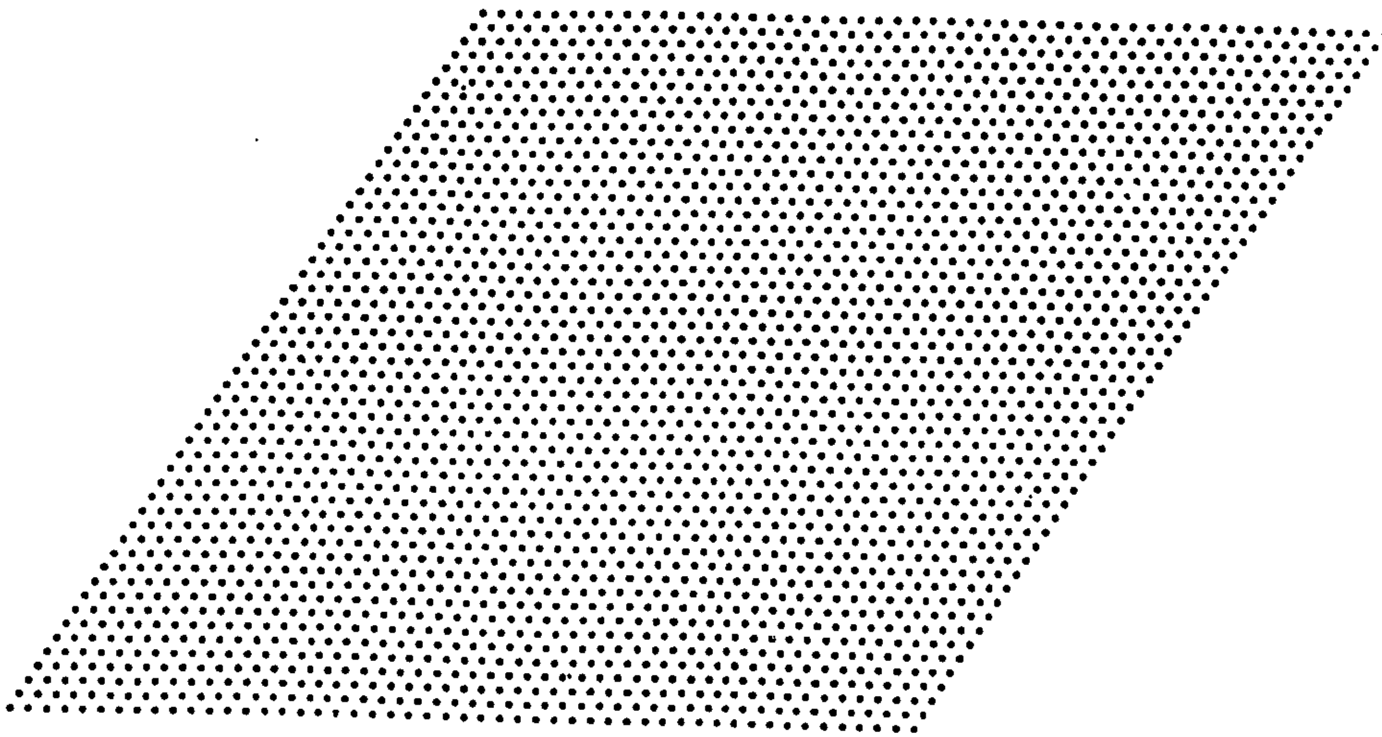

Figure 4.1: Initial Position of the atoms of xenon in 2D

4.2 Dimensionality of adsorbed gases:

Although all real adsorption systems are three-dimensional, it is in multilayer physisorption that this is most clearly evident. However, it is also true even if no multilayers are formed, provided that the atoms of the adsorbed species can be positioned at various distances from the adsorbing surface. The system can utilize this extra distance parameter so as to minimize its energy.

For this reason, the initial configuration of xenon atoms along the vertical coordinate z is used as;

$$z_i = z_0 + \Delta z \,(2\,\mathcal{R}_i - 1) \tag{4.1}$$

where,

$\mathcal{R}_i$	:$\mathcal{R}_i \in [0,1]$
z_0	: represents the position for the minimum of the energy of interaction of the atoms of xenon with the surface (111) of silver.
Δz	: is the radius of the potential wall for the potential represented in figure (2.2) for given temperatures.
i	: is the index of the atom
T	: is the temperature of the system

z_i represents the vertical position of the atom i of xenon above the plane of the substrate.

What we did, is that we created a disorder in our system around the position of minimum of potential energy. In the opposite case we found that the thermalisation is too slow.

The choise of Δz as function of temperature is given by:

$$\mathcal{V}(z) - \mathcal{V}(z_0) = \frac{1}{2}\,k_B\,T \tag{4.2}$$

Where $\mathcal{V}(z_0)$ represents the minimum potential energy and $\mathcal{V}(z)$ the potential energy from figure 2.2 in chapter 2. for the interaction of xenon atoms with the surface (111) of silver. T is the temperature.

The results of the use of this principal part is shown in figures 4.12, 4.13 and

4.14 in chapter 4.

4.3 Periodic boundary conditions:

To reduce the effect of the surface we impose periodic boundary conditions, the basis cell is identically repeated an infinite number of times.

In using periodic boundary conditions, we imagine that if a particle k is at $\mathbf{r}_k$, there is a set of image particles at positions $\mathbf{r}_k + \mathbf{t}_{mn}$, where $\mathbf{t}_{mn}$ is a superlattice Bravais vector, is equal to $\mathbf{t}_{mn} = m\mathbf{a} + n\mathbf{b}$, where $\mathbf{a}$ and $\mathbf{b}$ are primitive vectors of the cell, m and n are integers. In the computational implementation, if a particle crosses a surface of the basic cell it re-enters through the opposite wall with unchanged velocity as shown in figure(4.3). With these boundary conditions we have eliminated the surface and created a quasi-infinite volume to represent the macroscopic system more closely.

In a three dimensional system, any molecule in the gas above the surface is confined by reversing its velocity should it cross a plane at a height L_z above the surface. If L_z is sufficiently large, this reflecting boundary will not influence the behaviour of the adsorbed monolayer see figure(4.2).

Periodic boundary conditions have the advantage of eliminating edge effects, which can completely dominate the behavior of very small systems.

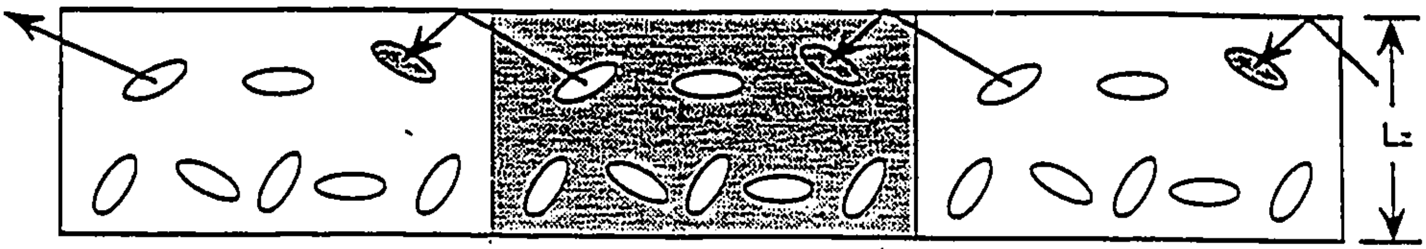

Figure 4.2: A three-dimensional periodic system. A side view of the box with the reflecting boundary at a height L_z

4.4 Constant Temperature:

The simplest procedure for simulating constant temperature takes the instantaneous temperature T_I defined by ;

$$\frac{3}{2} N\, k_B\, T_I = \sum_{k=1}^{N} \frac{\mathbf{p}_k^2}{2m} \tag{4.3}$$

to equal the desired temperature, the atomic velocities are renormalized at every time step so that the instantaneous mean Kinetic energy corresponds to the chosen temperature T.

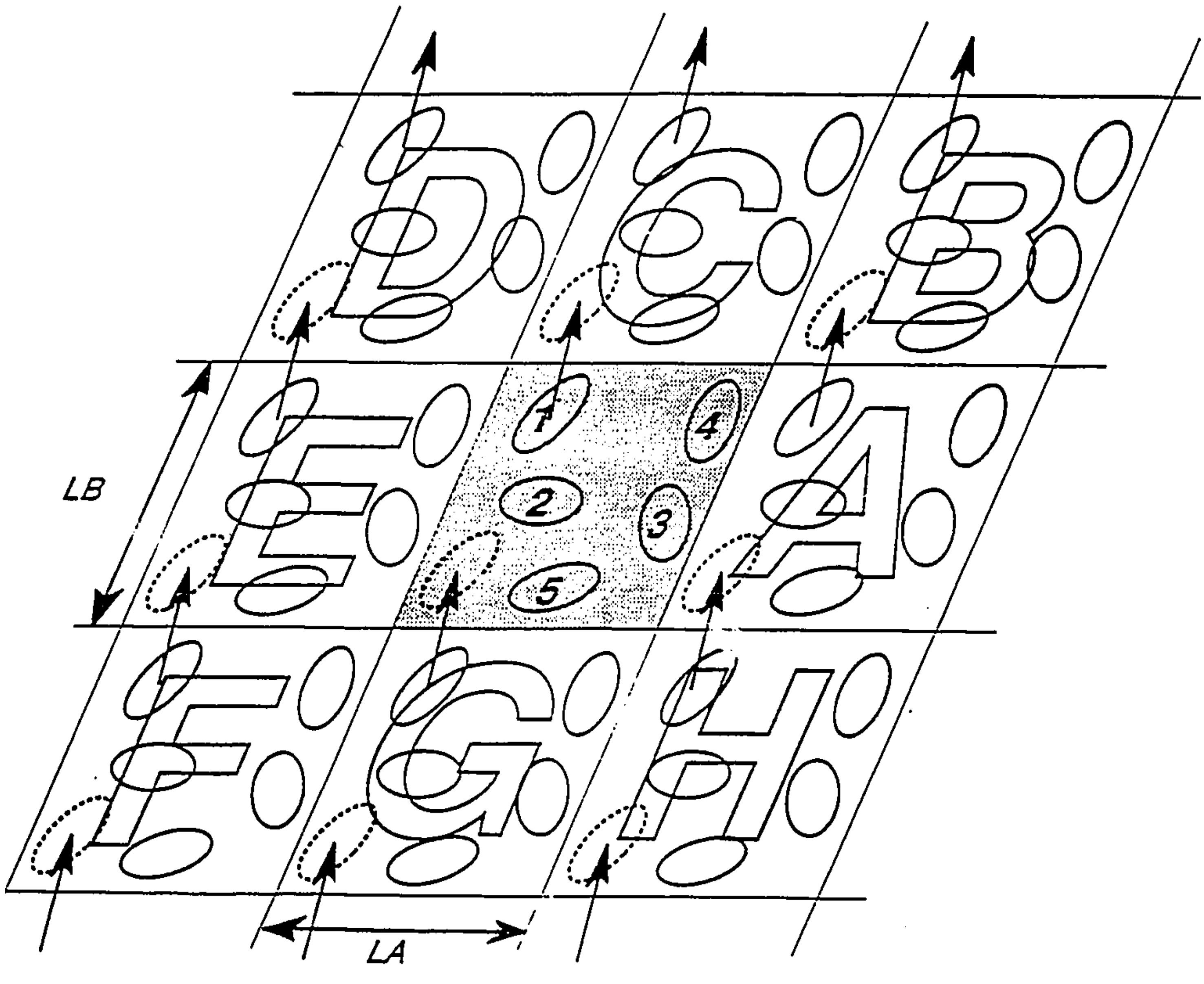

Figure 4.3: A two-dimensional periodic system. Molecules can enter and leave each cell across each of the four edges.

4.5 Iteration Constant Pressure:

The Xe floating monolayer was studied for a wide range of lattice parameters. However, because of contributions from the core region and inelastic behavior in general, the assembly relaxed at constant area will have an internal pressure P which is not equal to the external pressure P_E. To eliminate this difference in pressure, we introduce a Taylor expansion with respect to area or parameter change Δa in first order;

$$P \simeq P_E + \left(\frac{\partial P}{\partial a}\right)_{P=P_E} \Delta a \tag{4.4}$$

To achieve a constant-pressure we apply several iterations.
$\left(\frac{\partial P}{\partial a}\right)_{P=P_E}$ is given by the slope of the tangent at different parameters and perfect systems at pressure P_E from figure 4.4 .
We had to draw a series of isotherms to be able to find the pressure that could give us the right lattice-parameter at any temperature figure 4.4.
The pressure P will be given by ;

$$P = \frac{1}{2\mathcal{A}} \left(\sum_{k=1}^{N} \frac{1}{m} \mathbf{p}_k \, \mathbf{p}_k + \sum_{k=1}^{N} \sum_{k>l}^{N} \mathbf{r}_{kl} \, \mathbf{F}_{kl} \right) \tag{4.5}$$

In monolayer adatoms, the calculated pressure in the system will be represented by the surface pressure P. $\mathcal{A}$ represents the area for our system the Xe monolayer. Some results of the fluctuations of P in time are given at the end of this Chapter.

4.6 Method of Simulation Chosen:

The configuration energy for vacancy migration is obtained by holding an atom fixed at the position desired and allowing all other atoms to relax to their equi-

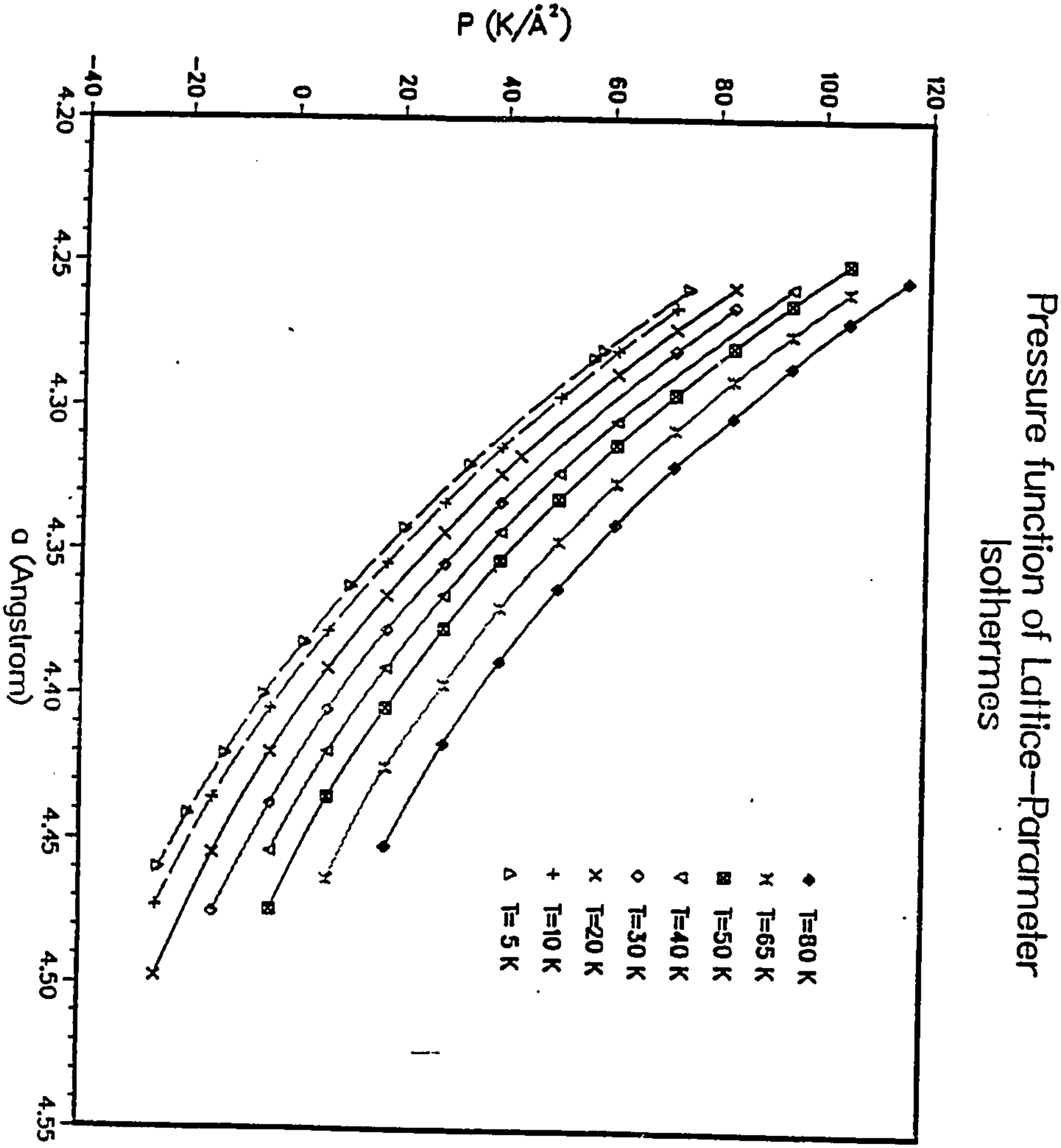

Figure 4.4: Pressure function of lattice-parameter for isotherms and Xe monolayer .

librium positions. The main concern was to maintain the atoms at the edge of the system fixed. If the atoms on the edge were not held at a fixed position, the system would return to its initial state. This procedure leads to a constant-area simulation.

The first molecular dynamics method we used was the combined method as described in section 3.3.6 of chapter 3. This method created an area fluctuation which caused irregularity in our calculations. However, this difficulty can be overcome by using the Damped force method as described in section 3.3.4 of chapter 3.

The next step was to maintain the same pressure for all the calculations . The iteration to reach constant pressure is described in section 4.4 of this chapter. In the three dimensional adsorbed monolayer , the scaling factor β was calculated in two parts β_{xy} and β_z,

$$\beta_{xy}{}^2 = \frac{2\,(N-1)\,k_B\,T_c/m}{\sum_{k=1}^{N}[\;\mathrm{v}'{}^2_{kx}(t) + \;\mathrm{v}'{}^2_{ky}(t)]} \tag{4.6}$$

and

$$\beta_z{}^2 = \frac{(N-1)\,k_B\,T_c/m}{\sum_{k=1}^{N}\;\mathrm{v}'{}^2_{kz}(t)} \tag{4.7}$$

The reason was that there is a low dynamic coupling between the two-dimensional direction and the third dimensional direction. The interparticle forces have only a small vertical component. For the same scaling factor, the thermalisation is too slow. We inforce β_{xy} and β_z to maintain the thermalisation .

For the interaction energy between two vacancies, we do not have any conditions concerning the atoms at the edges of our system. In this case we used the combined method as described in section 3.3.6 of chapter 3. [38], where constant temperature and constant pressure are provided by the algorithm itself.

We introduced the change of the factor M given in equation (3.33) , to reduce the fluctuations of our quantities after a number of time steps and to approach faster the thermal equilibrium. See figures 4.27 to 4.32.

4.6.1 Some Results of Molecular Dynamics Simulation:

The values of the pressure have to be divided by $10^{+16} k_B$ to get the unit $K/\mathring{A}^2$, this rearrangement will be taken in consideration when we do energy calculations.

The potential energies in figures showing the evolution of energy with time are energies by atom. And the value used for calculations of these quantities are average values.

I use the name "**Activated atom**" to designate the atom which makes the positional interchange with the vacancy.

The average calculations of the variables of interest are done only after the systems reach thermal equilibrium.

By NVT Molecular Dynamics Simulation:

Figure 4.5 : Presents the time-dependence of the pressure at temperature equal to $5K$ for a two dimensional xenon monolayer. The fluctuations in pressure are smaller than those given by a constant pressure simulation method. We did not present the first 500 steps because the fluctuations are higher and they present a non-physical state. We see from this figure that the fluctuations in pressure change very quickly at a higher frequency than in the perfect system, this is due to the maximum potential energy created by the position of the activated atom.

Figure 4.6 : Presents the evolution of the average pressure during NVT molecular dynamics simulation for a two dimensional monolayer at temperature of $5K$. Each coordinates of the average pressure is given by the sum of the precedent instantaneous pressures divided by the number of steps. From this figure , we see that the average value of pressure between 3000 and 5000 steps is constant.

Figure 4.7 : Presents the evolution of the potential energy per atom during NVT molecular dynamics simulations for a two dimensional xenon monolayer. The fluctuations of the energy are similar to those for pressure (fig. 4.5). The variation of the potential energy is equivalent to the variation of the second term of the pressure see equation 4.5. The first term of the pressure (Equ. 4.5) is constant for a given temperature because the total kinetic energy of the system is maintained constant.

Figure 4.8 : Presents the evolution of average energy during NVT molecular dynamics simulation for a two dimensional monolayer, at temperature of $5K$. Same procedure as average pressure (Fig. 4.6).

Figure 4.9 : Presents the positions of atoms in a final configuration for a two dimensional xenon monolayer, for the case where the activated atom is at the midpoint between the two sites for the positional interchange. And this done at temperature of $5K$ after 5000 time steps.

Figure 4.10 : Presents the evolution of the pressure during NVT molecular dynamics simulations for a perfect two dimensional xenon monolayer. The amplitude of the fluctuations is greater than the ones in figure 4.5 due to the

higher temperature, $50K$. The other difference from fig 4.5 is that the pressure at a lower frequency of oscillations than in the system where the activated atom is held fixed.

Figure 4.11 : Presents the evolution of the potential energy per atom for a two dimensional ideal monolayer, which is similar to figure 4.10 case. The fluctuations of the energy are similar to those in fig. 4.10, the explanation is the same as the one given for fig. 4.7. This is also done at a temperature of $50K$.

Figure 4.12 : Presents the time-dependance of the part of the potential energy which represent the Xe-Xe interaction in a three dimensional adsorbed monolayer for 1000 time steps. The thermal equilibrium is reached after 3000 time steps.

Figure 4.13 : Presents the time-dependence of the part of the potential energy which represent the Xe-substrate interaction in a three dimensional adsorbed monolayer for 1000 time steps. The energy fluctuations have been minimized, by applying the principle explained in section 4.2 of chapter 4. This variation look like a phenomenon of beats with decrease in amplitude.

Figure 4.14 : Presents the time-dependence of the total potential energy which is the sum of the results given by figures 4.12 and 4.13 at a temperature of $5K$. The thermal equilibrium is reached after 3000 time steps.

Figure 4.15 : Presents the evolution of the two dimensional pressure during NVT molecular dynamics simulation for a three dimensional adsorbed monolayer. The fluctuations are similar to those for the potential energy given for

the interaction Xe-Xe in fig. 4.12 because, in pressure calculations, we do not introduce the part of the interaction of the monolayer with the substrate. The pressure represents a surface pressure instead of volumic pressure for the adsorbed monolayer of xenon atoms on silver(111).

Figure 4.16 : Represents the final configuration of the positions of the atoms of xenon in the (X, Y) plane for the three dimensional adsorbed monolayer at temperature of $5K$ after 10000 time steps in the equilibrium state.

Figure 4.17 : Shows the variation of the vertical coordinate Z of the activated atom which makes the positional interchange with the vacancy after a displacement of $-0.3a$ (a : lattice parameter) along the X axis, the coordinate along Y axis is kept the same. This shows that the activated atom fluctuate along Z axis and settles into the equilibrium state after about 2500 time steps which is about $2.5\ 10^{-11}$ seconds.

Figure 4.18 : Similar to fig. 4.17 but for a different displacement which is in this case $-0.5a$ along X axis, in this case we see that the equilibrium state is reached just after 1000 time steps.

Figure 4.19 : Evolution of the pressure for 3D adsorbed monolayer, for perfect system at temperature of $50K$. The difference from fig 4.15 is that the amplitude of fluctuations of the pressure is greater due to the increase in temperature.

Figure 4.20 : Similar to fig. 4.14 but for a different temperature which is in this case $50K$. The difference is the same as the one mentioned for the evolution of pressure in fig. 4.19 .

By NPT combined Molecular Dynamics Simulation:

Figure 4.21 : The evolution of the pressure during NPT molecular dynamics with M kept constant for the whole simulation. The choice of its value 10^{-20} is explained in section 3.3.5 of chapter 3. We chose this particular value of M because the statistic fluctuations due to the volume are not visible, but the thermic fluctuations are presented by a high frequency of oscillations. This is done for two dimensional monolayer at temperature of $5K$. The fluctuations are greater compared to the case of constant volume. The other difference from the fluctuations in constant volume is that the fluctuations are homogeneous because there is no coupling to a third degree of freedom.

Figure 4.22 : Presents the evolution of average pressure for the pressure given in fig. 4.21, the thermal equilibium is reached quickly, this is due to the increase in frequency of oscillations, result of the choise of M value. The procedure of average pressure canculations is the same as the one given for figure 4.6.

Figure 4.23 : Presents the evolution of the lattice-paramete for 5000 time steps. The variation is similar to the evolution of pressure given in figure 4.21. The amplitude of oscillations is small.

Figure 4.24 : Presents the evolution of the average of lattice-parameter for the lattice parameter variation given in fig. 4.23. The equilibrium state is reached quickly.

Figure 4.25 : Presents the evolution of the potential energy per atom . Same case as figure 4.21, similar result as the ones given in fig. 4.21 .

Figure 4.26 : Presents the evolution of the average of the potential energy per atom . Same case as figure 4.21, and same result as given in fig. 4.22 .

figure 4.27 : Describes the evolution of pressure in time. For a two dimensional xenon monolayer at temperatureof 10K. The difference from figure 4.21 is that after a number of time steps when the equilibrium is reached, we change the value of M to a value smaller from the first value. From 10^{-20} to 10^{-22}, we see immediatly the effects of this change, a decrease in amplitude of fluctuations and increase of their frequency. This method brings the system very quickly to an equilibrium state with more precision in average calculations.

Figure 4.28 : Presents the variation of the average pressure after the change of M is made. The effects of the change of M gives an average value stable.

Figure 4.29 : Presents the evolution of the lattice parameter, same case as figure 4.27. The results of fluctuations are similar as the case of figure 4.27. we see that after the value of M is changed, the variations in lattice parameter are very small, where the fluctuations in volume have less effects to reach the thermic equilibium.

Figure 4.30 : Describes the variation of the average lattice-parameter after the change of M is made. The effects of the change of M gives a very good precision of the average value of a.

Figure 4.31 : Presents the evolution of the potential energy. Same case as figure 4.27, the results of fluctuations are similar as the case of figure 4.27.

Figure 4.32 : Presents the evolution of the average energy. Same case as figure 4.27, The results of fluctuations are similar as the case of figure 4.28.

Figure 4.33 : Final configuration of the position of the atoms at temperature of $10K$ for a two dimensional xenon monolayer with two vacancies.

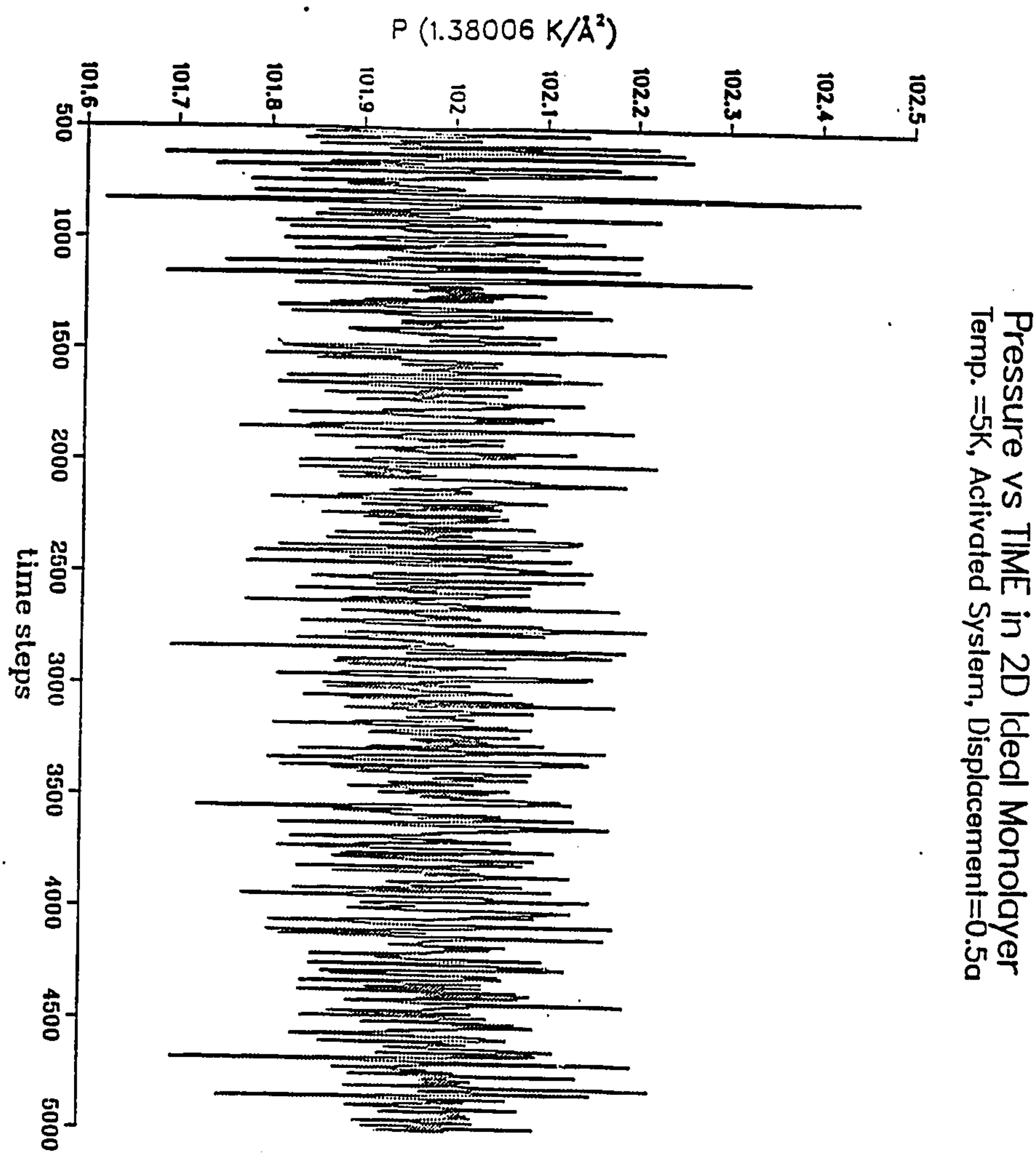

Figure 4.5: Evolution of the pressure during NVT molecular dynamics simulation for 2D ideal monolayer, activated system, displacement =0.5a (a:Lattice-parameter) at T=5K. $\Delta t = 2\,10^{-14}$ seconds.

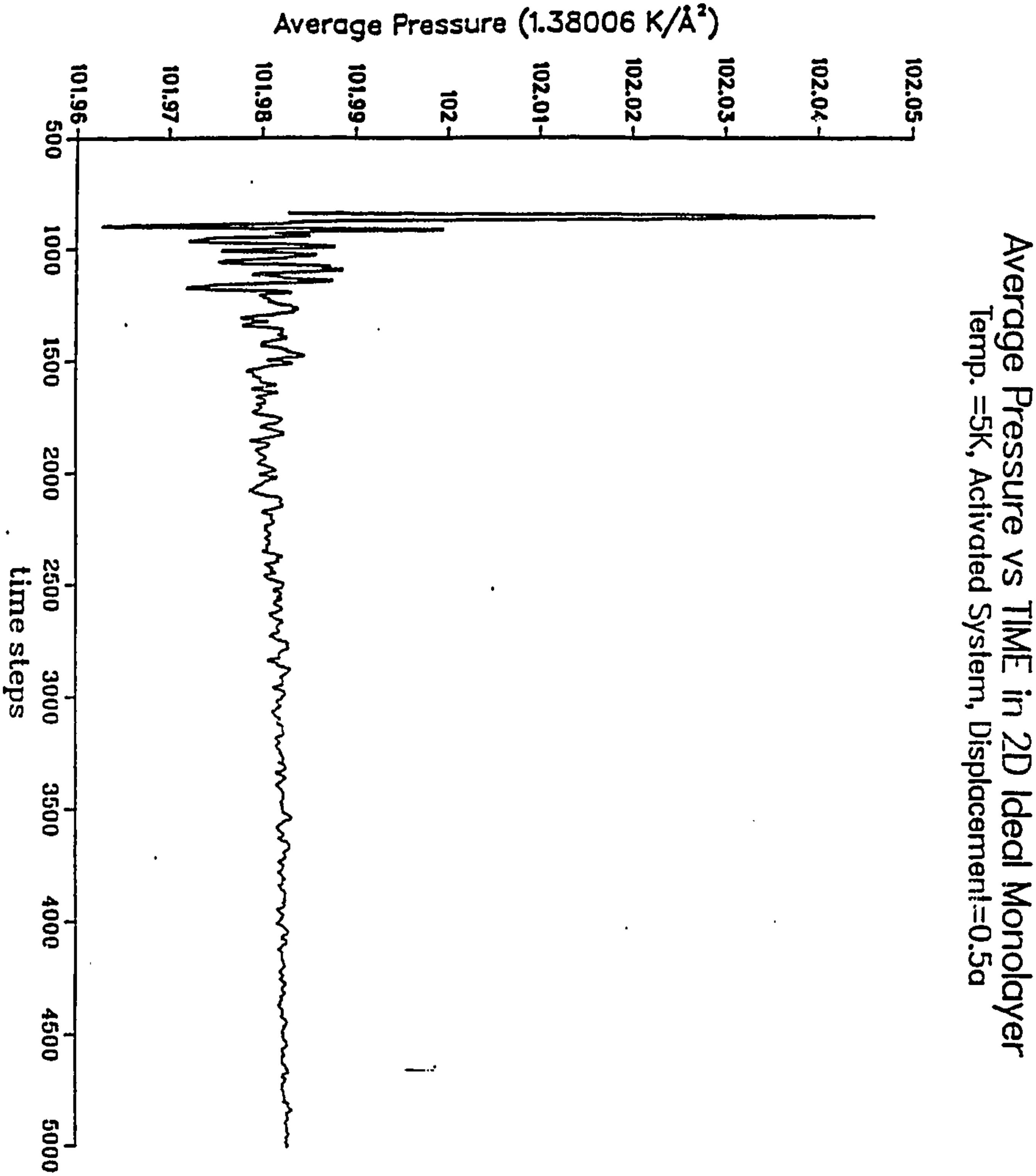

Figure 4.6: Evolution of the average pressure during NVT molecular dynamics simulation for 2D ideal monolayer , activated system, displacement =0.5a at T=5K, $\Delta t = 2\,10^{-14}$ seconds.

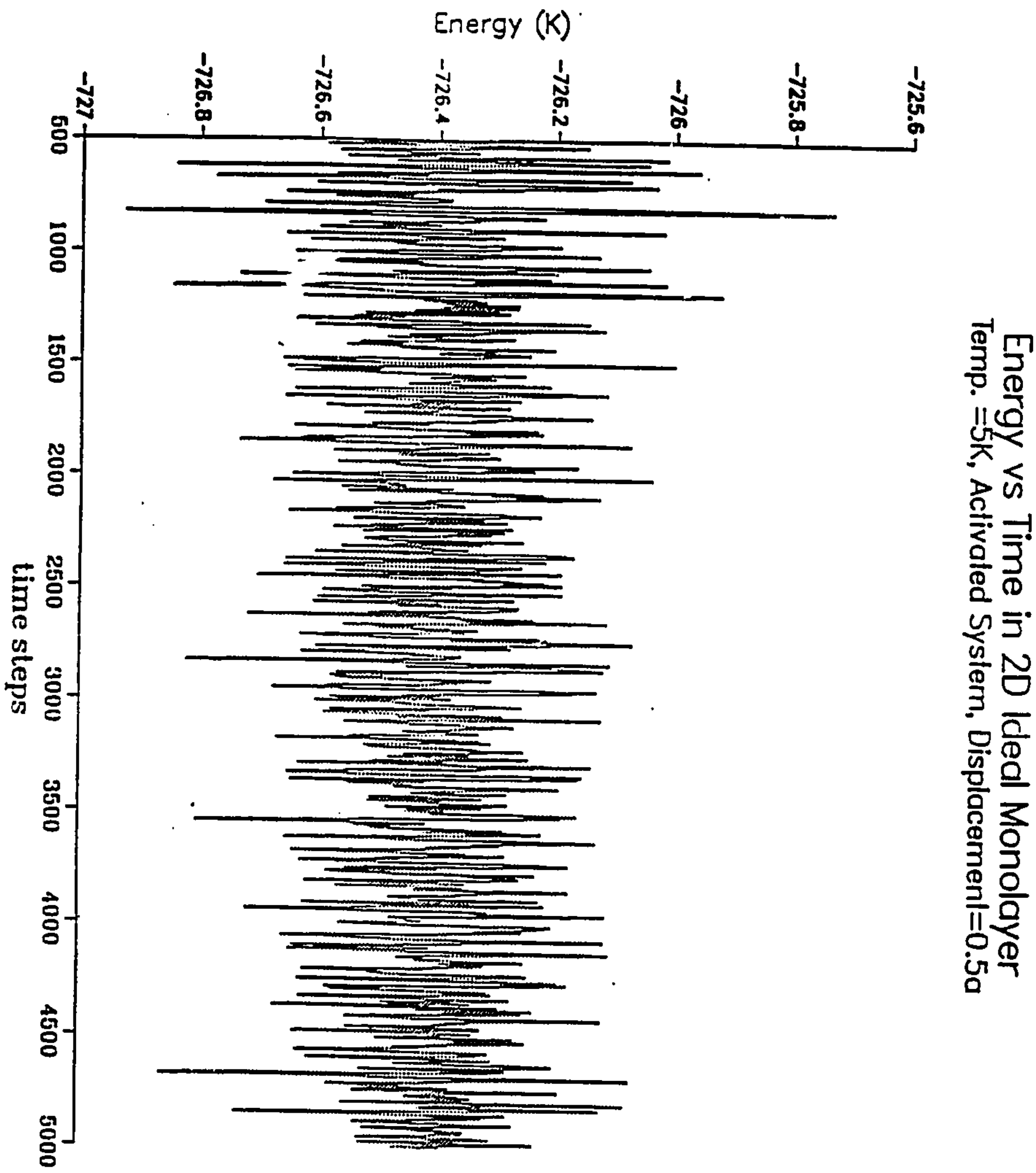

Figure 4.7: Evolution of the energy during NVT molecular dynamics simulation for 2D ideal monolayer, activated system, displacement =0.5a at T=5K between 500 and 5000 MD steps. $\Delta t = 2\ 10^{-14}$ seconds.

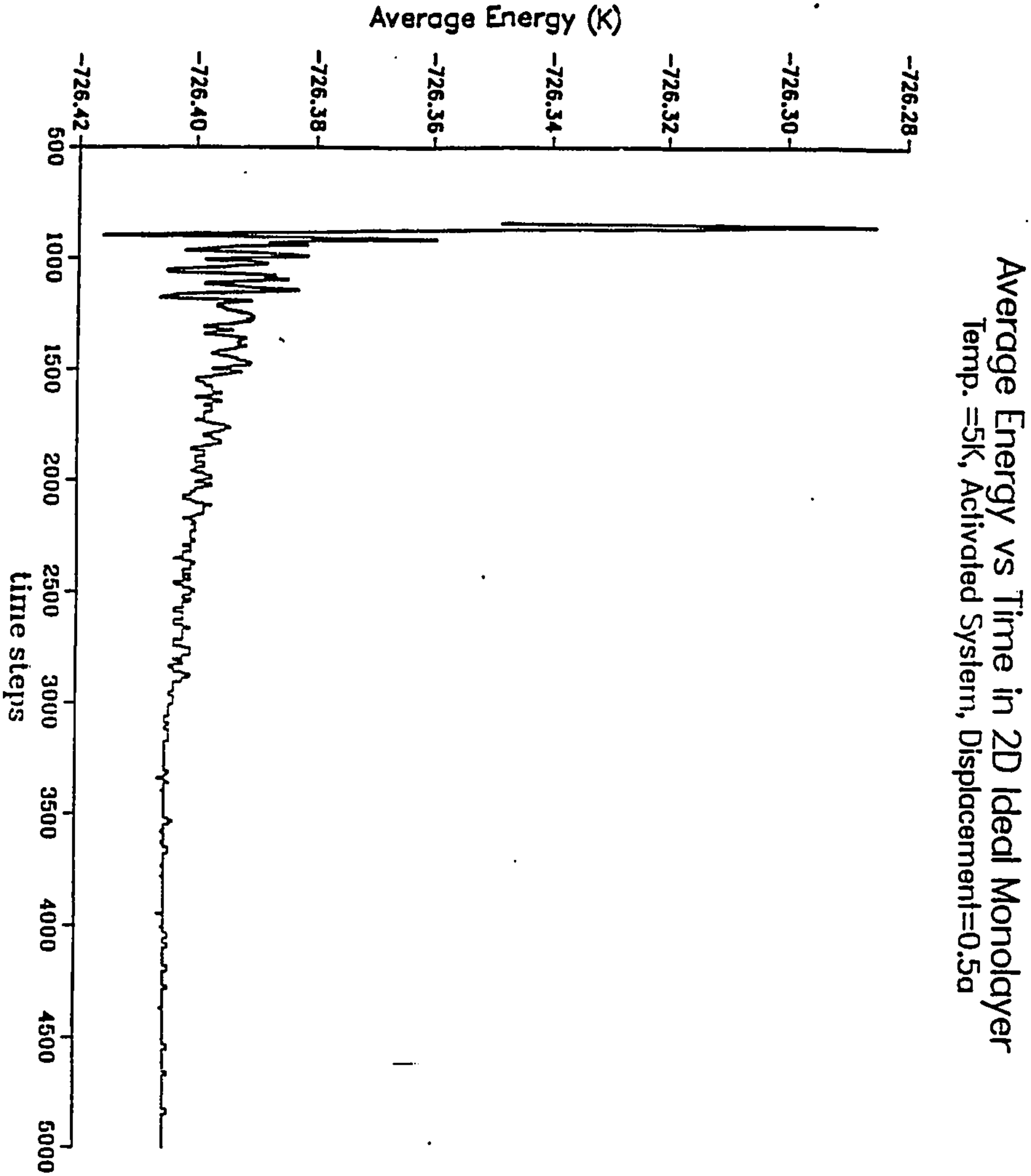

Figure 4.8: Evolution of the average energy during NVT molecular dynamics simulation for 2D ideal monolayer, activated system, displacement =0.5a at T=5K between 500 and 5000 MD steps, $\Delta t = 2\,10^{-14}$ seconds.

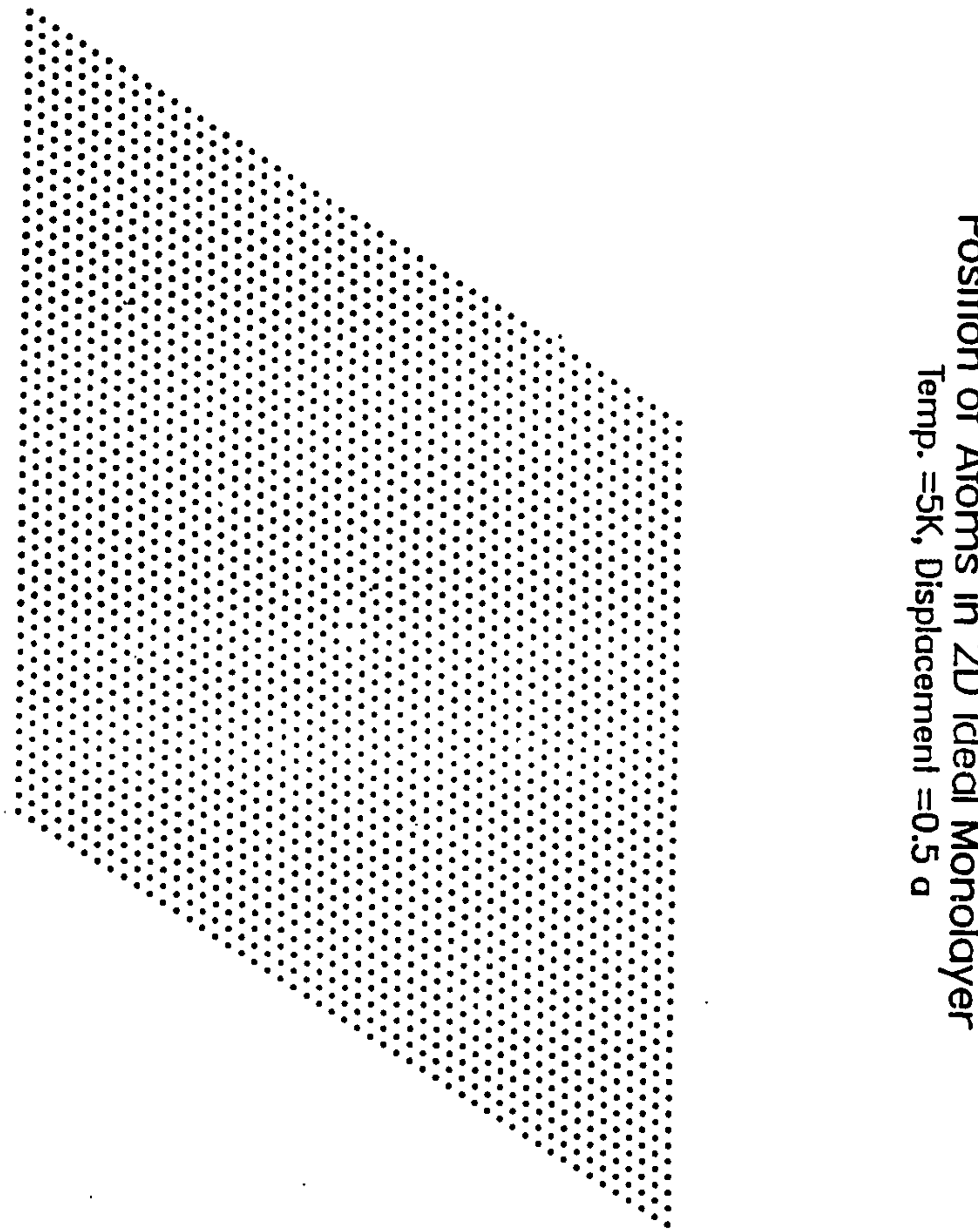

Figure 4.9: The position of atoms in final configuration when the atom is activated, displacement=0.5a (a:Lattice-parameter) at t= 5K, after 5000 MD steps.

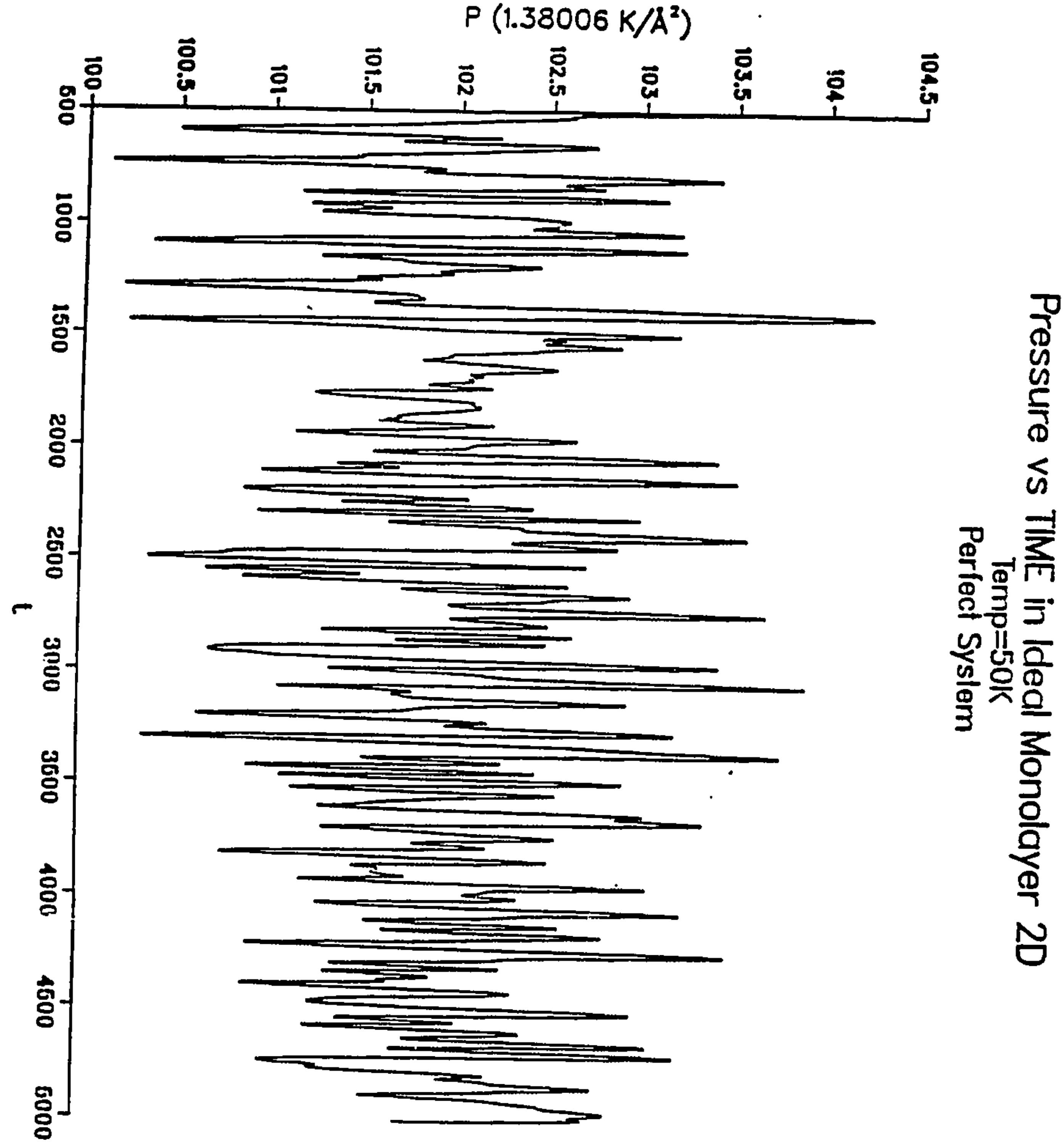

Figure 4.10: Evolution of the pressure during NVT molecular dynamics simulation for 2D ideal monolayer . perfect system at T=50K, $\Delta t = 210^{-14}$ seconds.

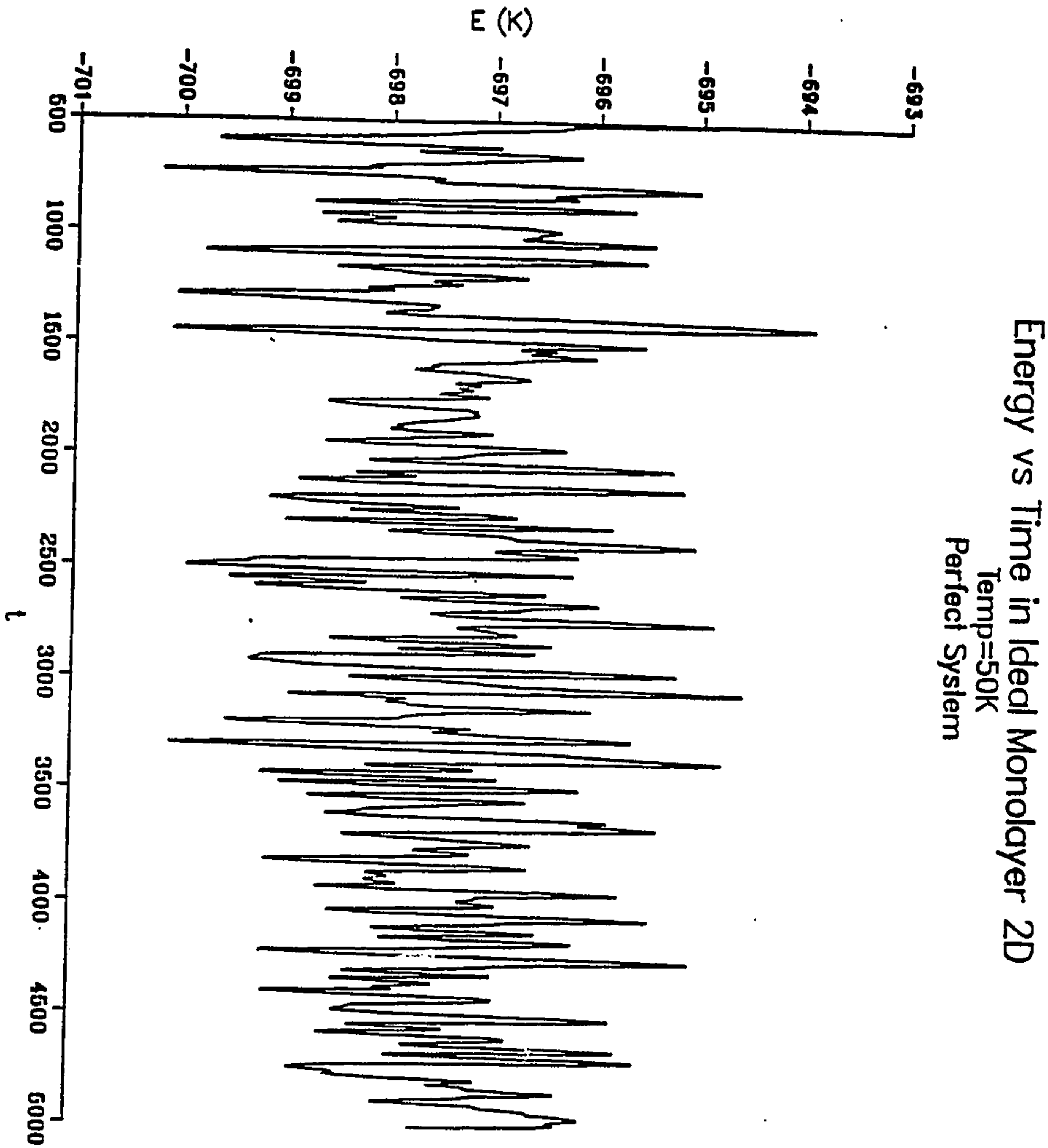

Figure 4.11: Evolution of the energy during NVT molecular dynamics simulation for 2D ideal monolayer, perfect system at T=50K between 500 and 5000 MD steps, $\Delta t = 2\ 10^{-14}$ seconds.

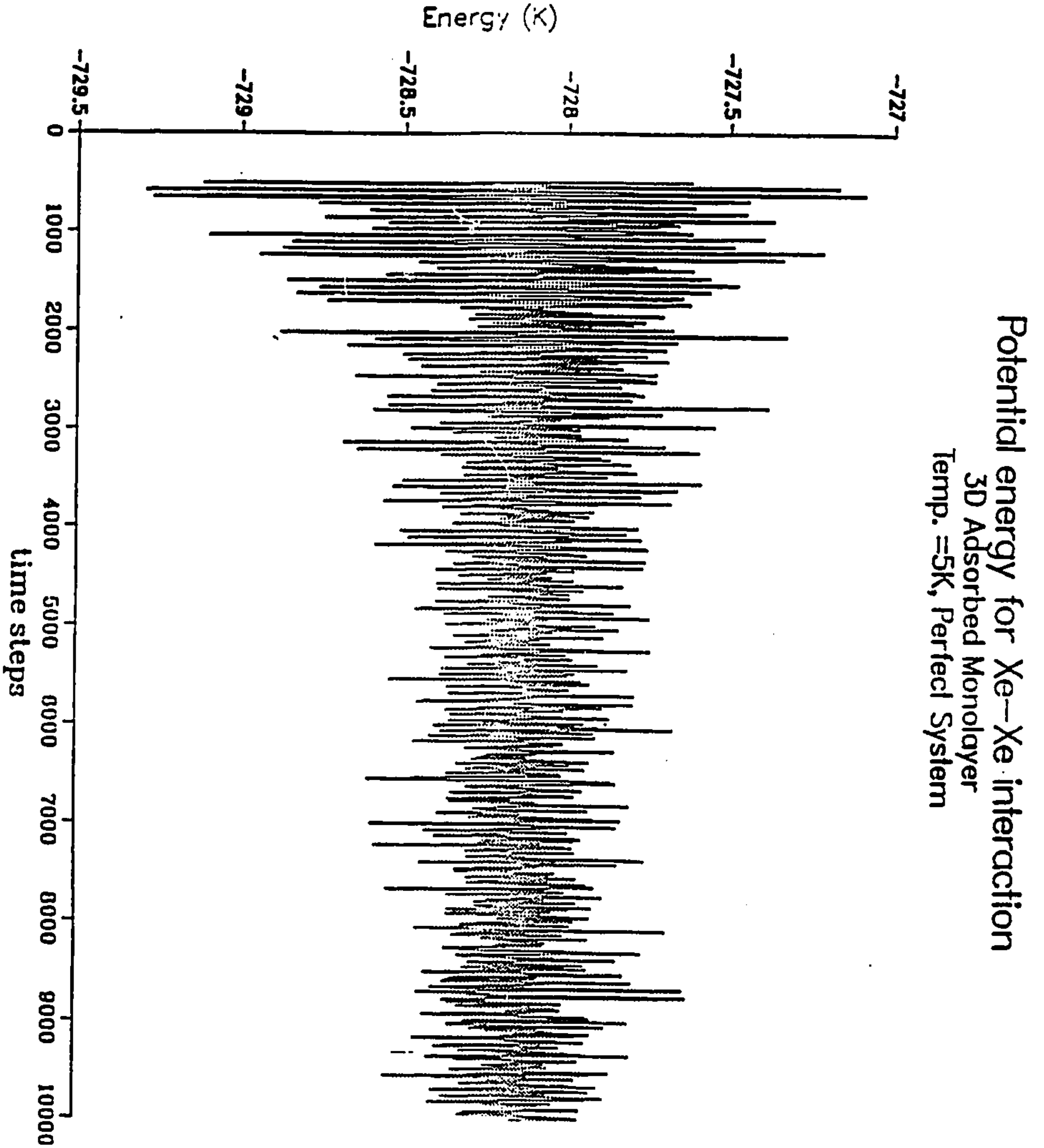

Figure 4.12: The potential energy for Xe-Xe interaction as a function of time steps 1-10000, which represent the first term in the equation (2.1), $\Delta t = 10^{-14}$ seconds.

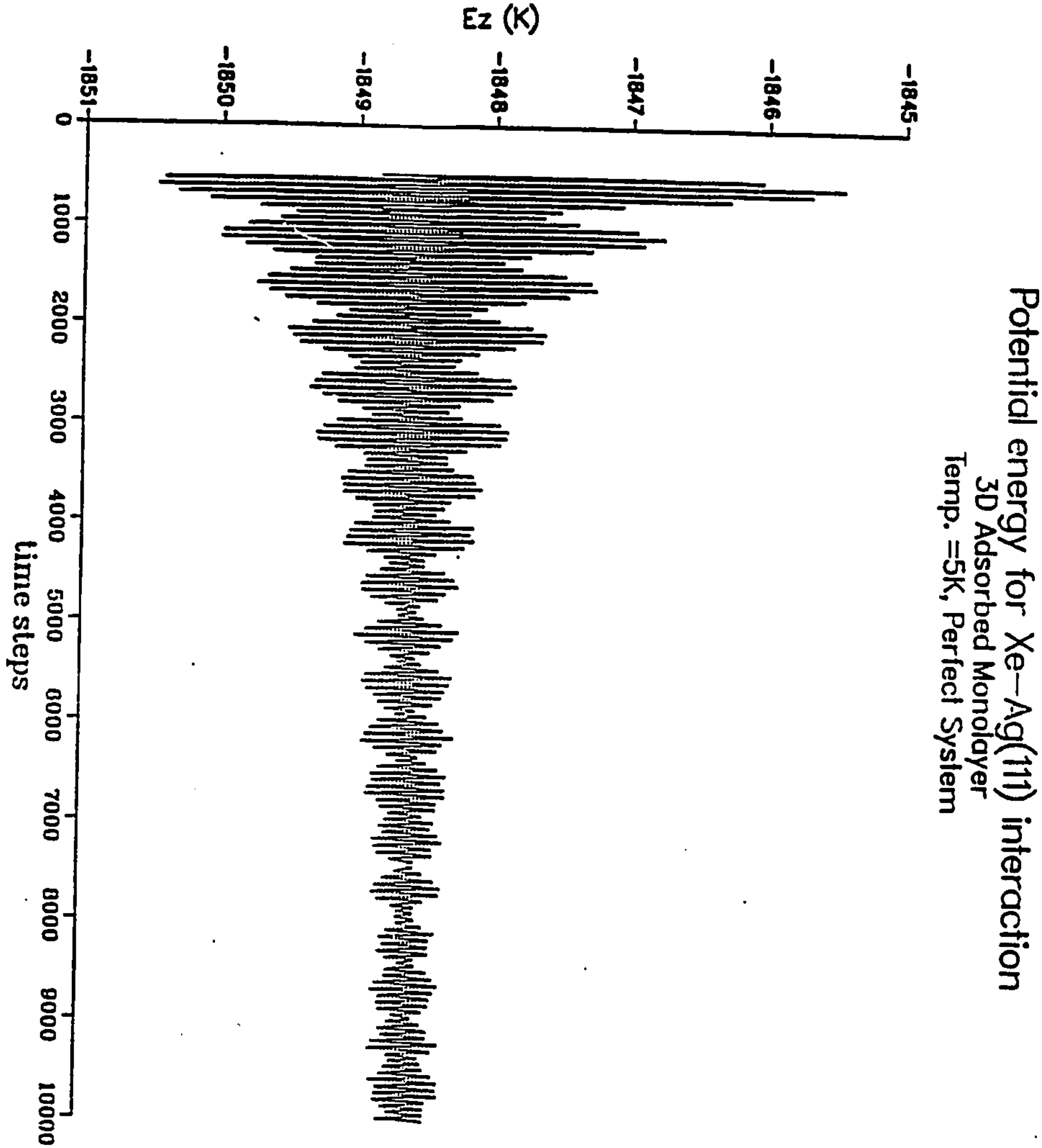

Figure 4.13: the potential energy Xe-substrate interaction as a function of time steps 1-10000, which represent the second term in the equation (2.1), $\Delta t = 10^{-14}$ seconds.

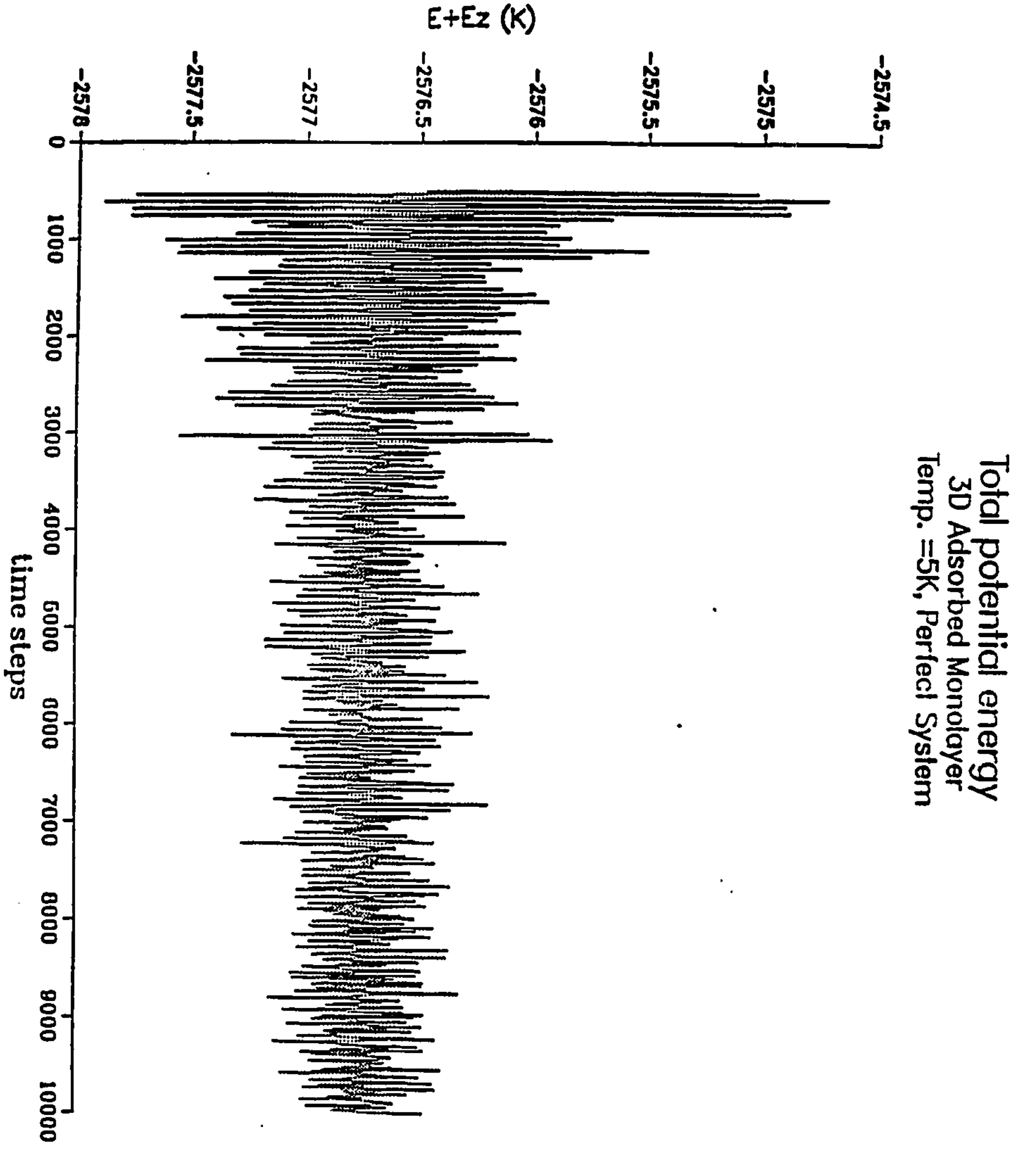

Figure 4.14: The total potential energy as a function of time steps 1-10000. represents U. $\Delta t = 10^{[illegible]}$ seconds.

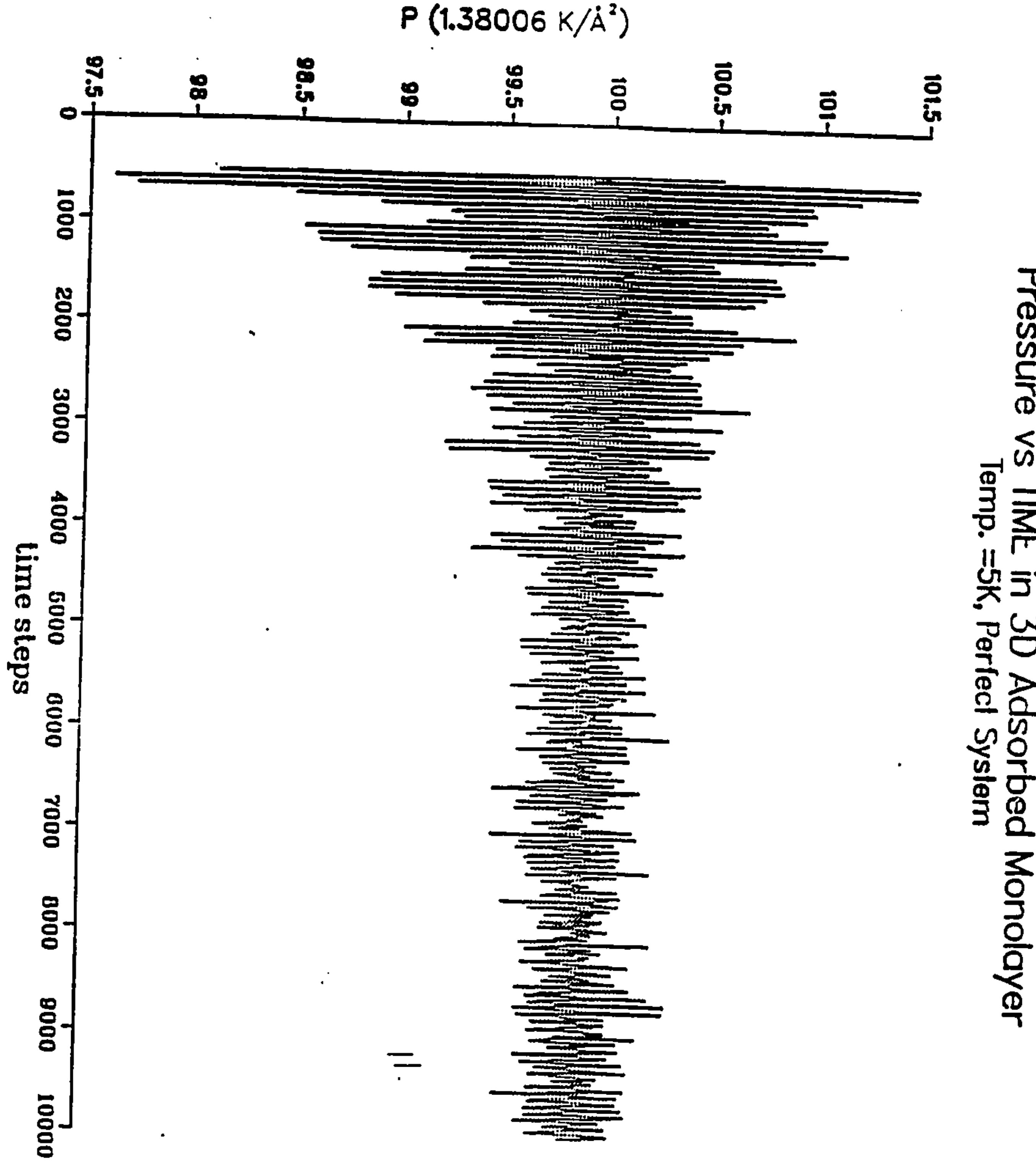

Figure 4.15: Evolution of the pressure during NVT molecular dynamics simulation for 3D adsorbed monolayer , perfect system at T=5K, $\Delta t = 10^{-14}$ seconds.

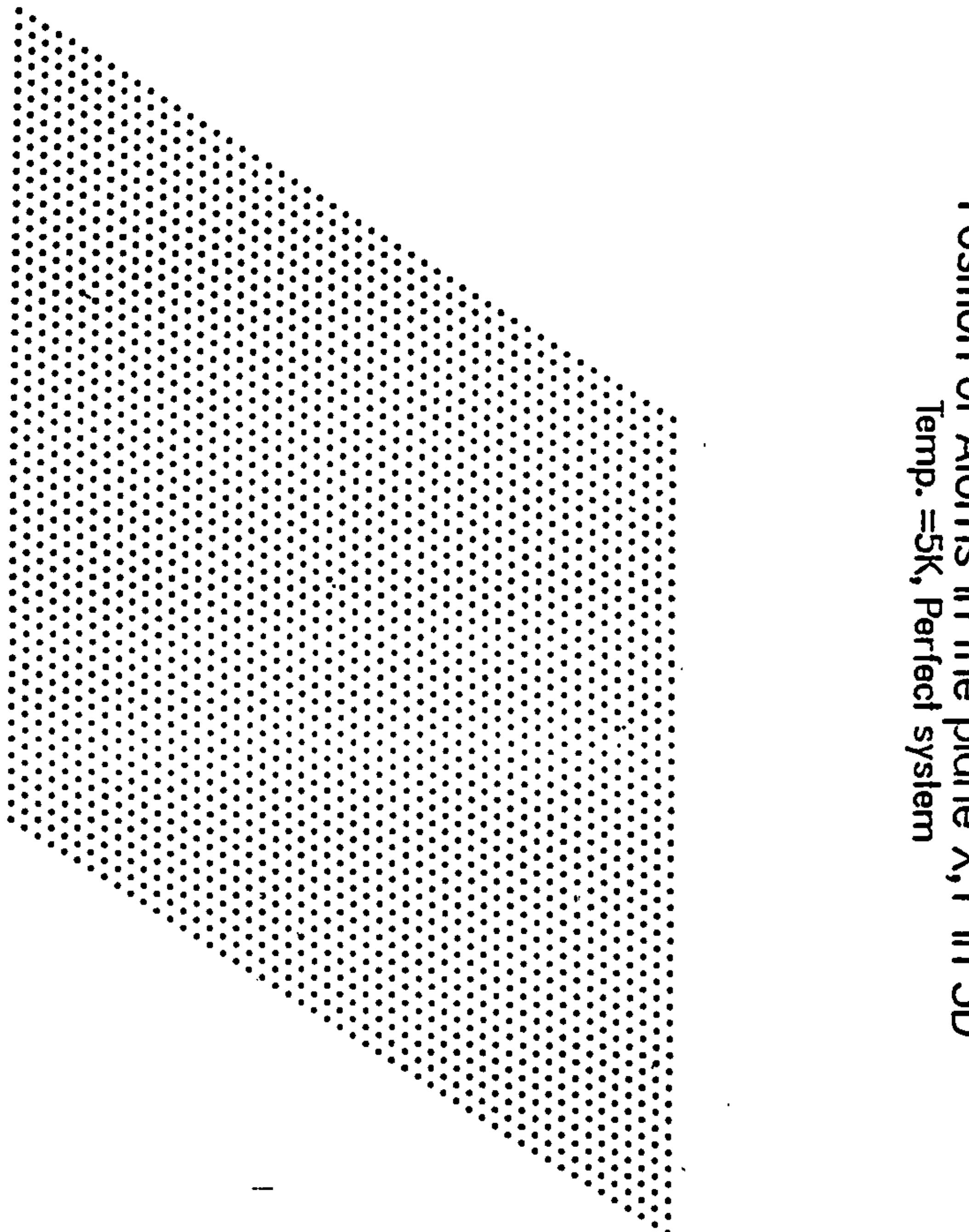

Figure 4.16: Final configuration of the position of atoms in the plan X and Y, for Xe 3D adsorbed monolayer at T=5K, perfect system , after 10000 time steps, $\Delta t = 10^{-14}$ seconds.

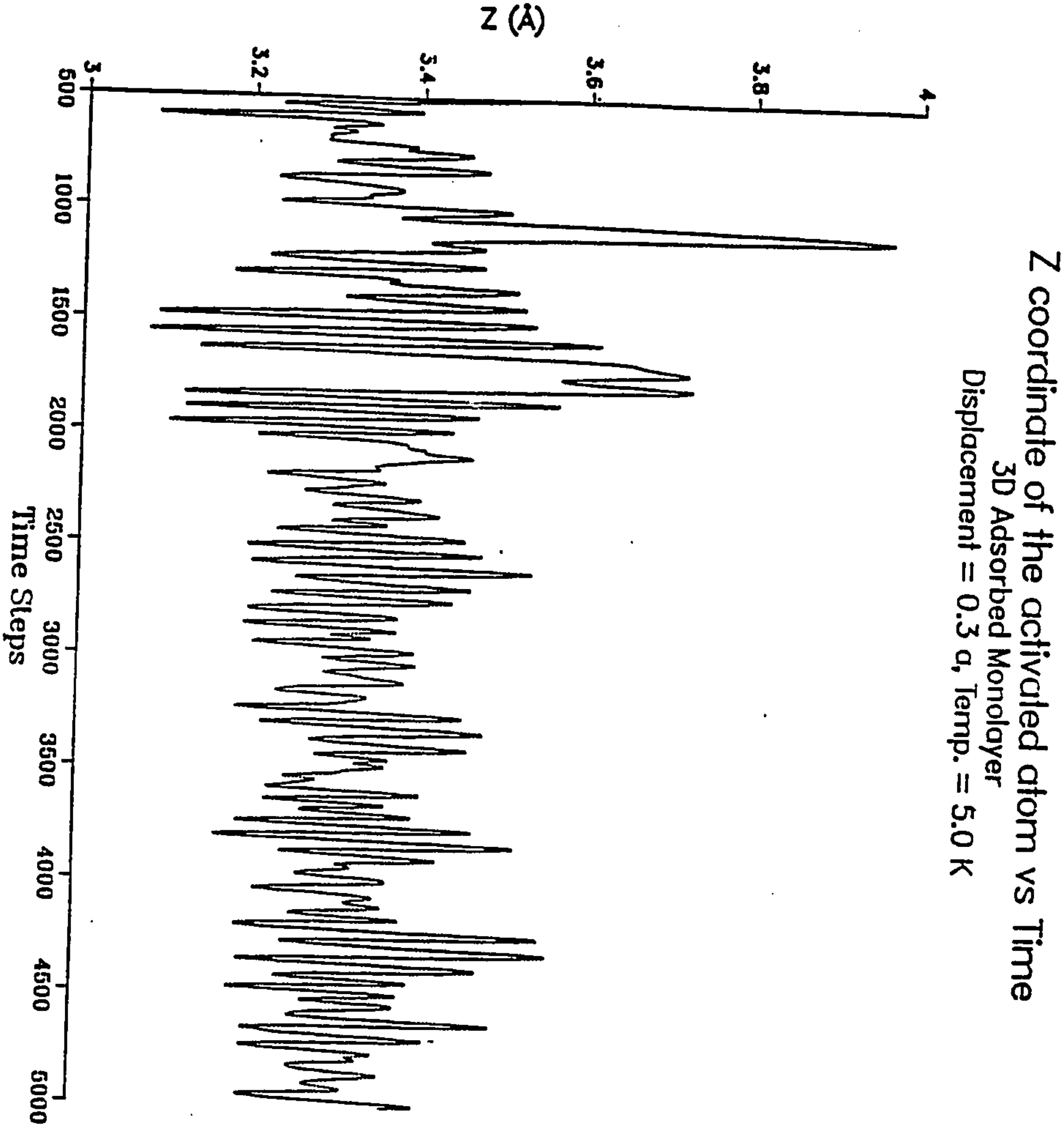

Figure 4.17: Evolution of Z coordinate of the activated atom, displacement= 0.3a (a:Lattice-parameter), at T=5K, $\Delta t = 10^{-14}$ seconds.

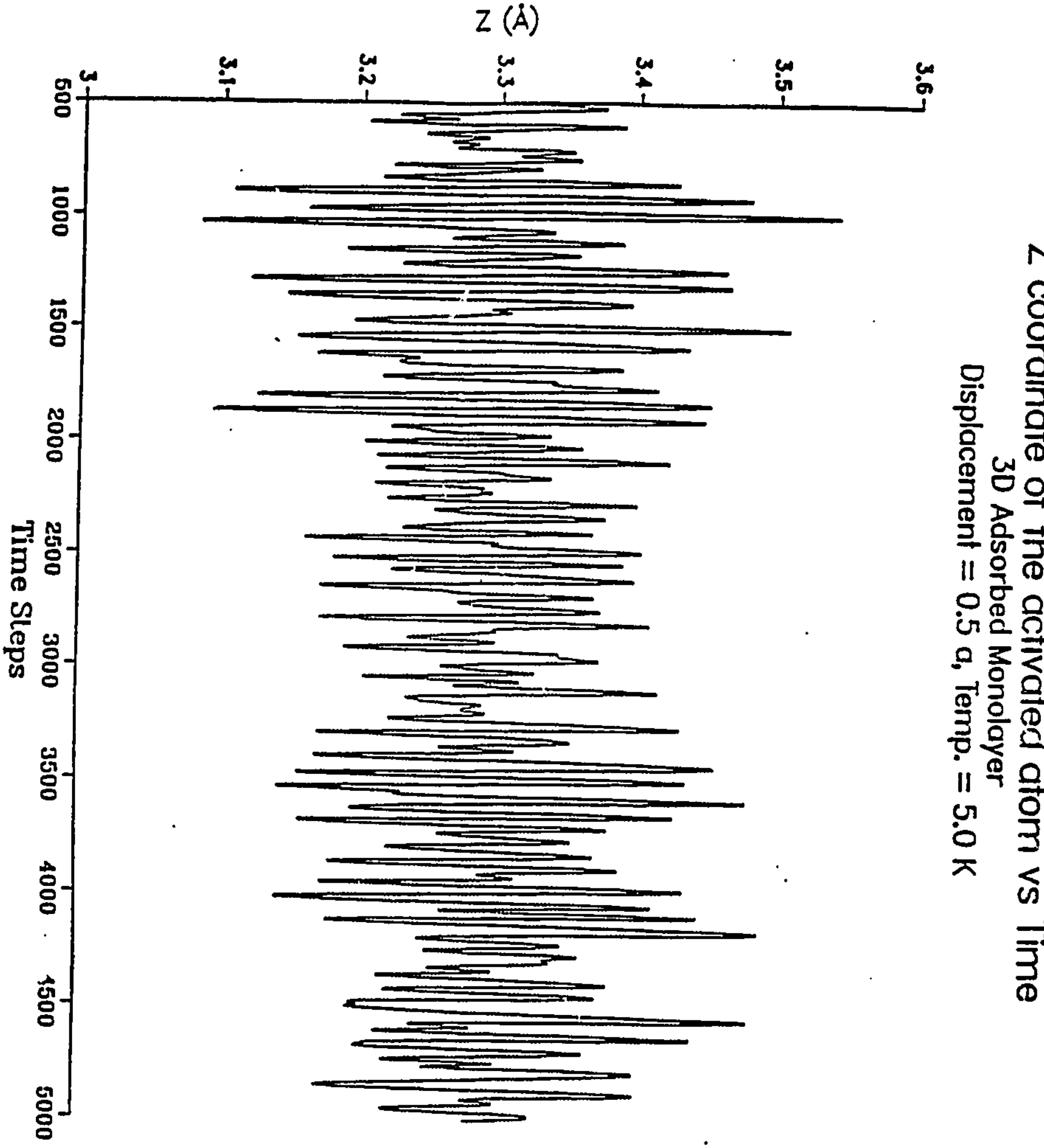

Figure 4.18: Evolution of Z coordinate of the activated atom, displacement= 0.5a (a:Lattice-parameter). at T=5K. $\Delta t = 10^{-14}$ seconds.

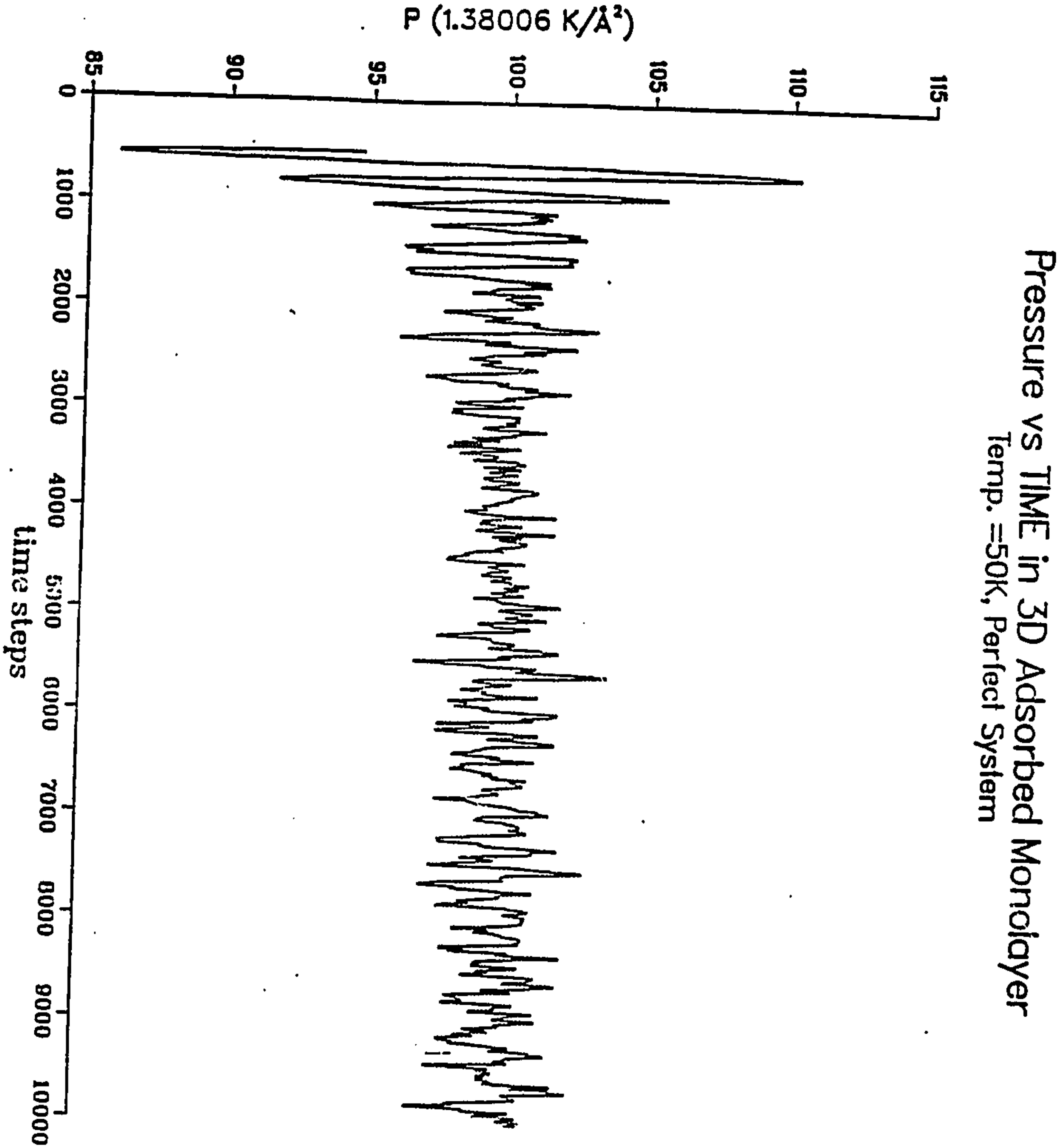

Figure 4.19: Evolution of the pressure during NVT molecular dynamics simulation for 3D adsorbed monolayer , perfect system at T=50K, $\Delta t = 10^{-14}$ seconds.

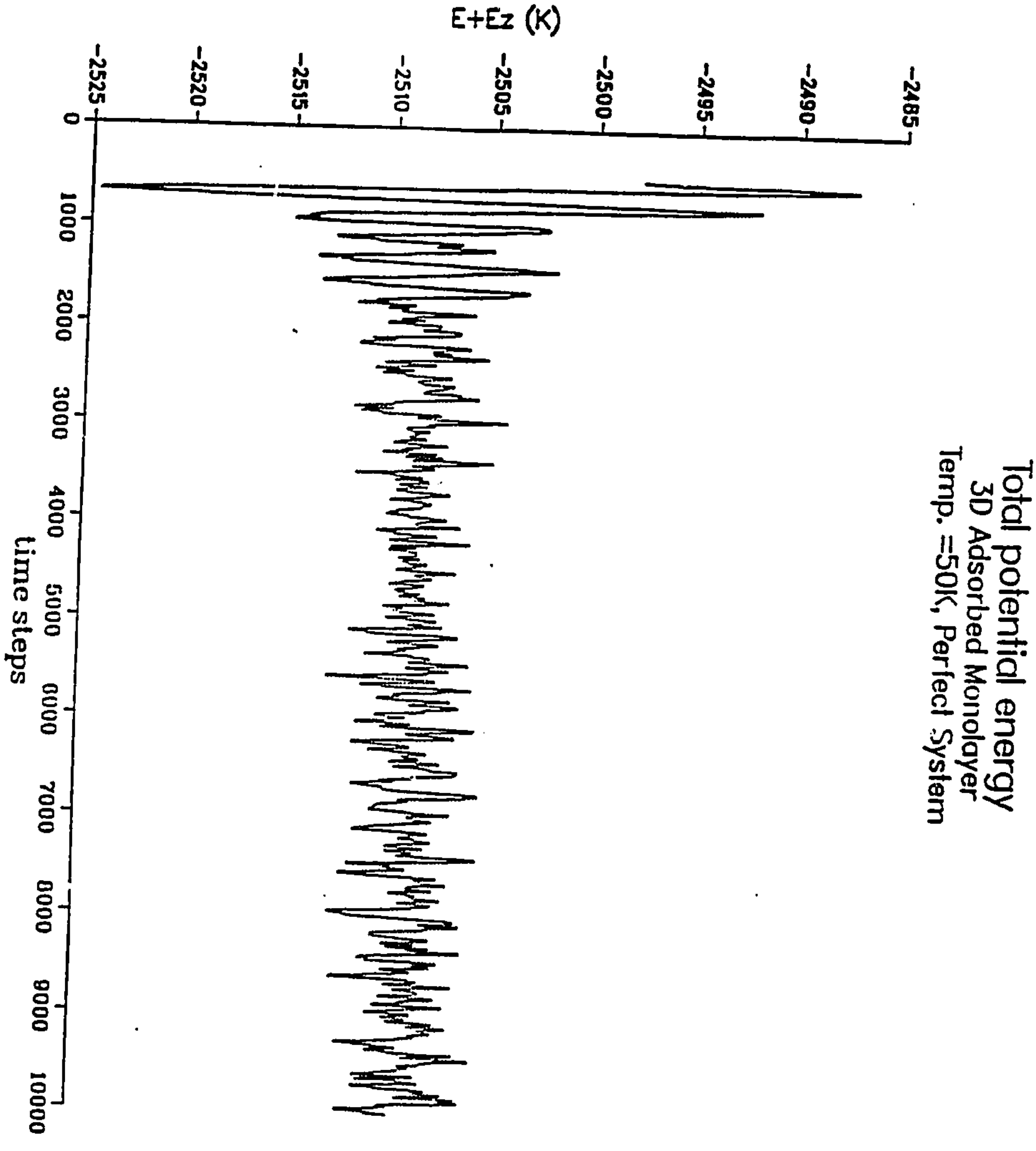

Figure 4.20: The total potential energy as a function of time steps 1-10000, at T= 50K, $\Delta t = 10^{-14}$ seconds.

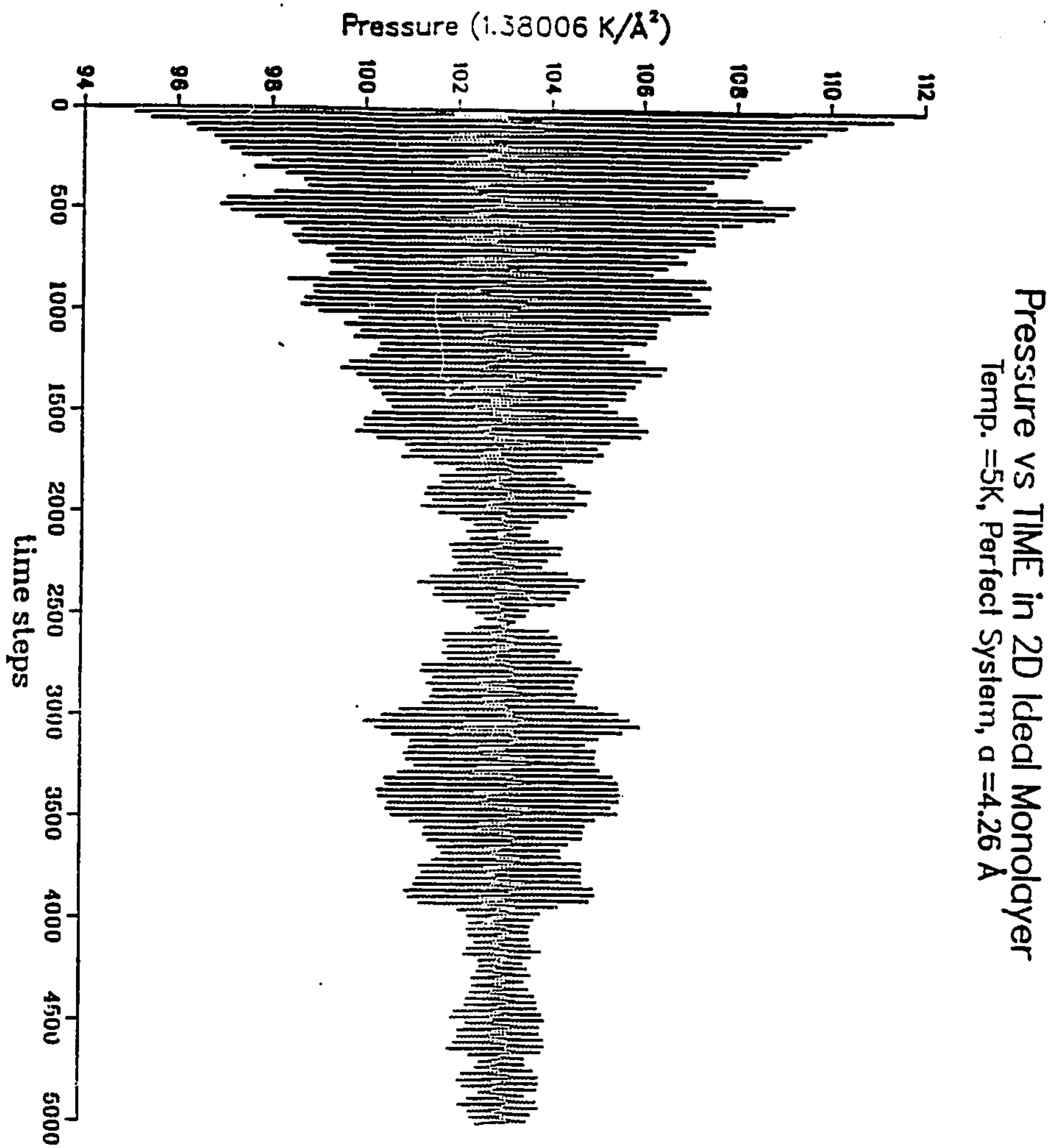

Figure 4.21: Evolution of the pressure during NPT molecular dynamics without changing the value of M, in 2D at $T = 5.0$K and $\Delta t = 2\ 10^{-14}$ seconds.

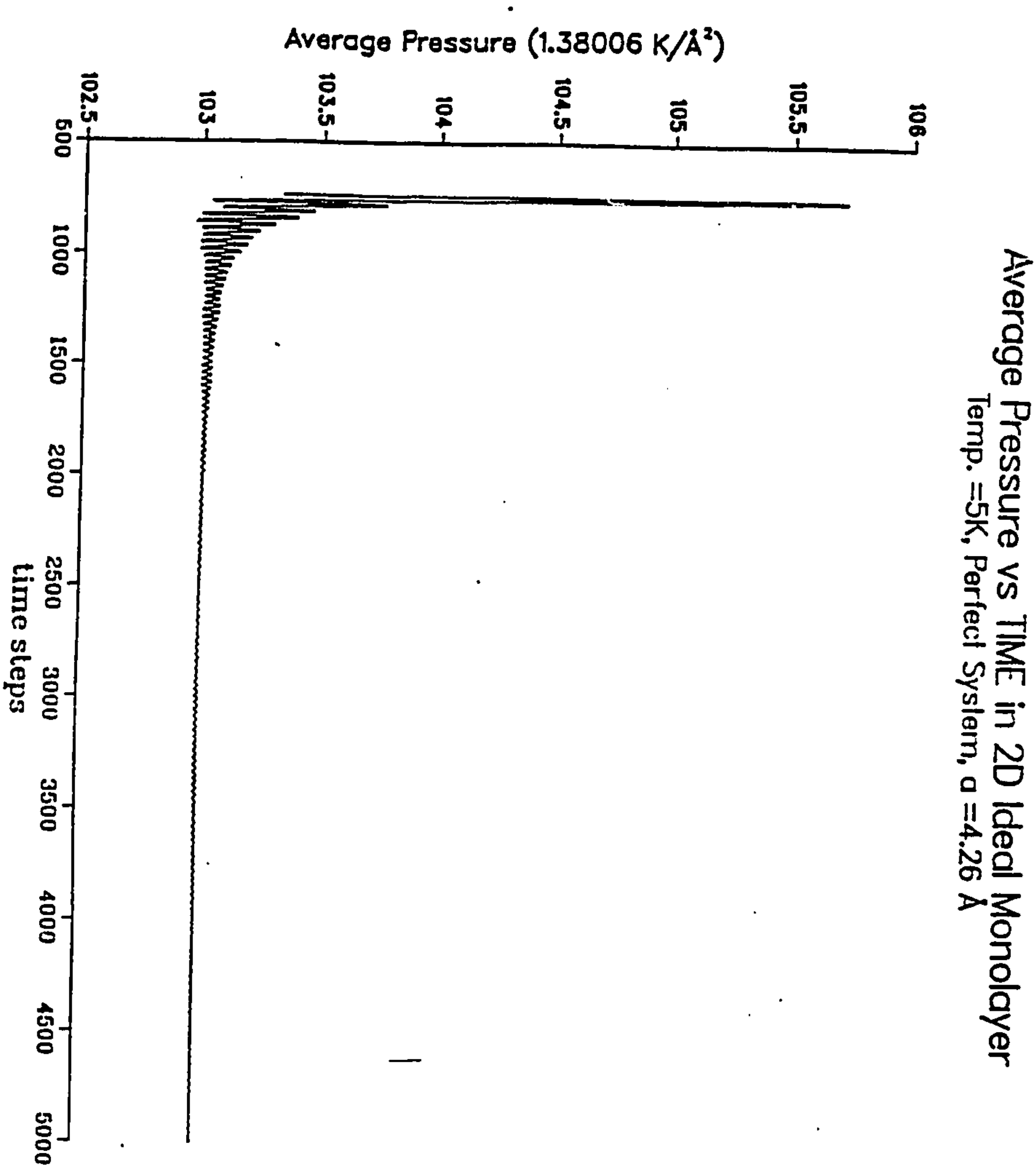

Figure 4.22: Evolution of the average pressure during NPT molecular dynamics without changing the value of M, in 2D at T=5.0K and $\Delta t = 2\,10^{-14}$ seconds.

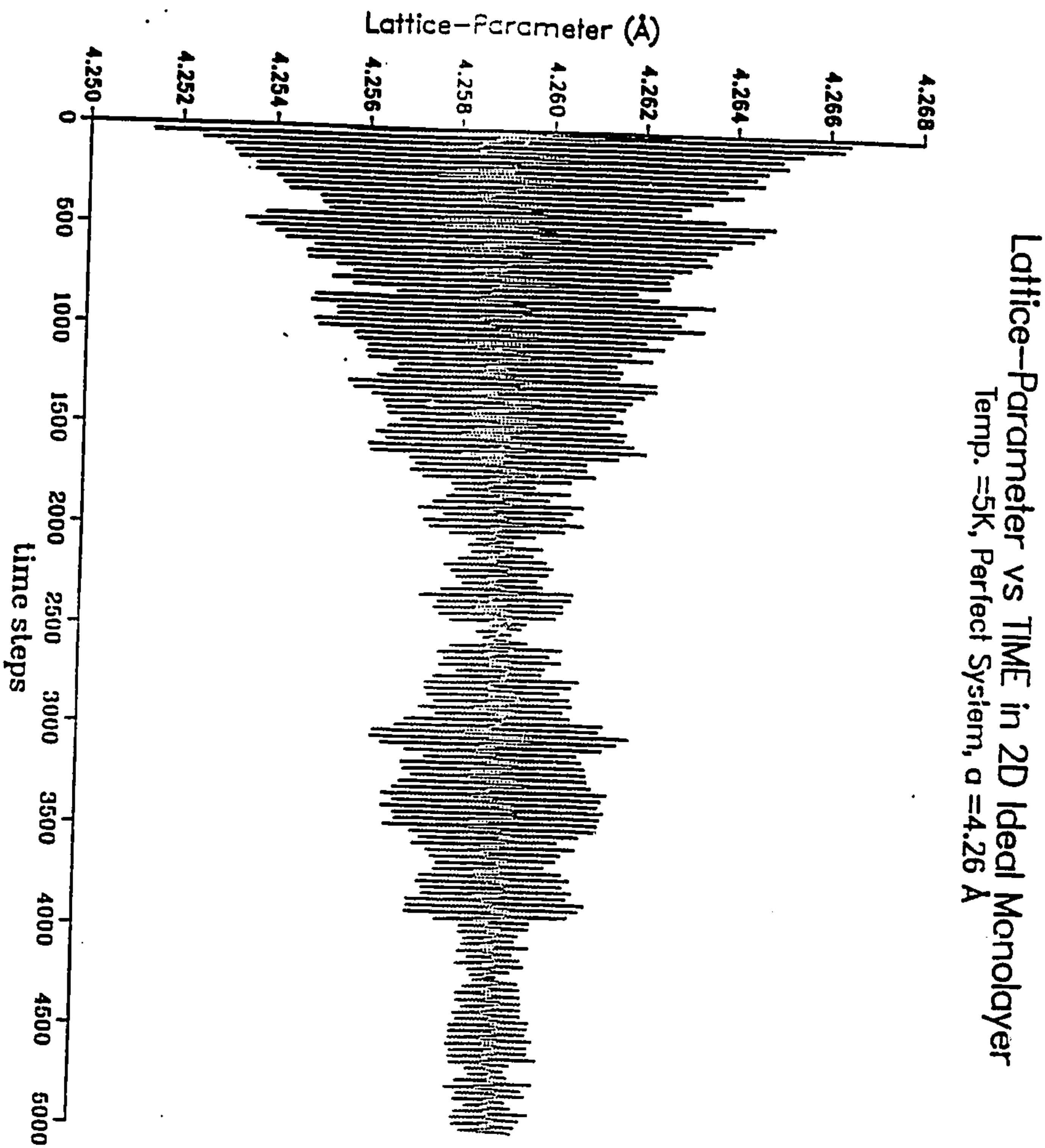

Figure 4.23: Evolution of the lattice parameter during NPT molecular dynamics without changing the value of M. in 2D at T=5.0K and $\Delta t = 2\ 10^{-14}$ seconds.

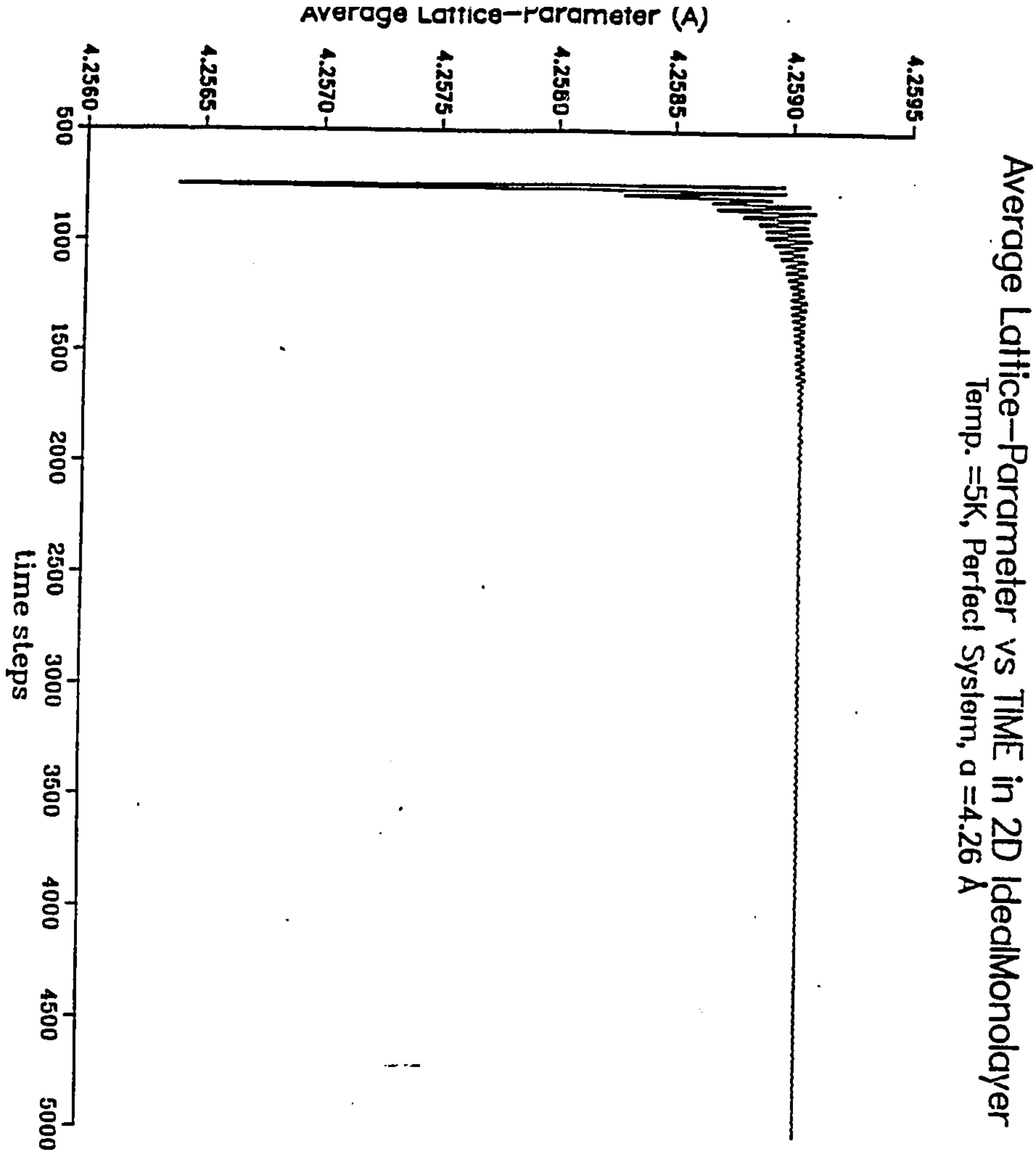

Figure 4.24: Evolution of the average parameter during NPT molecular dynamics without changing the value of M, in 2D at T=5.0K and $\Delta t = 2\ 10^{-14}$ seconds.

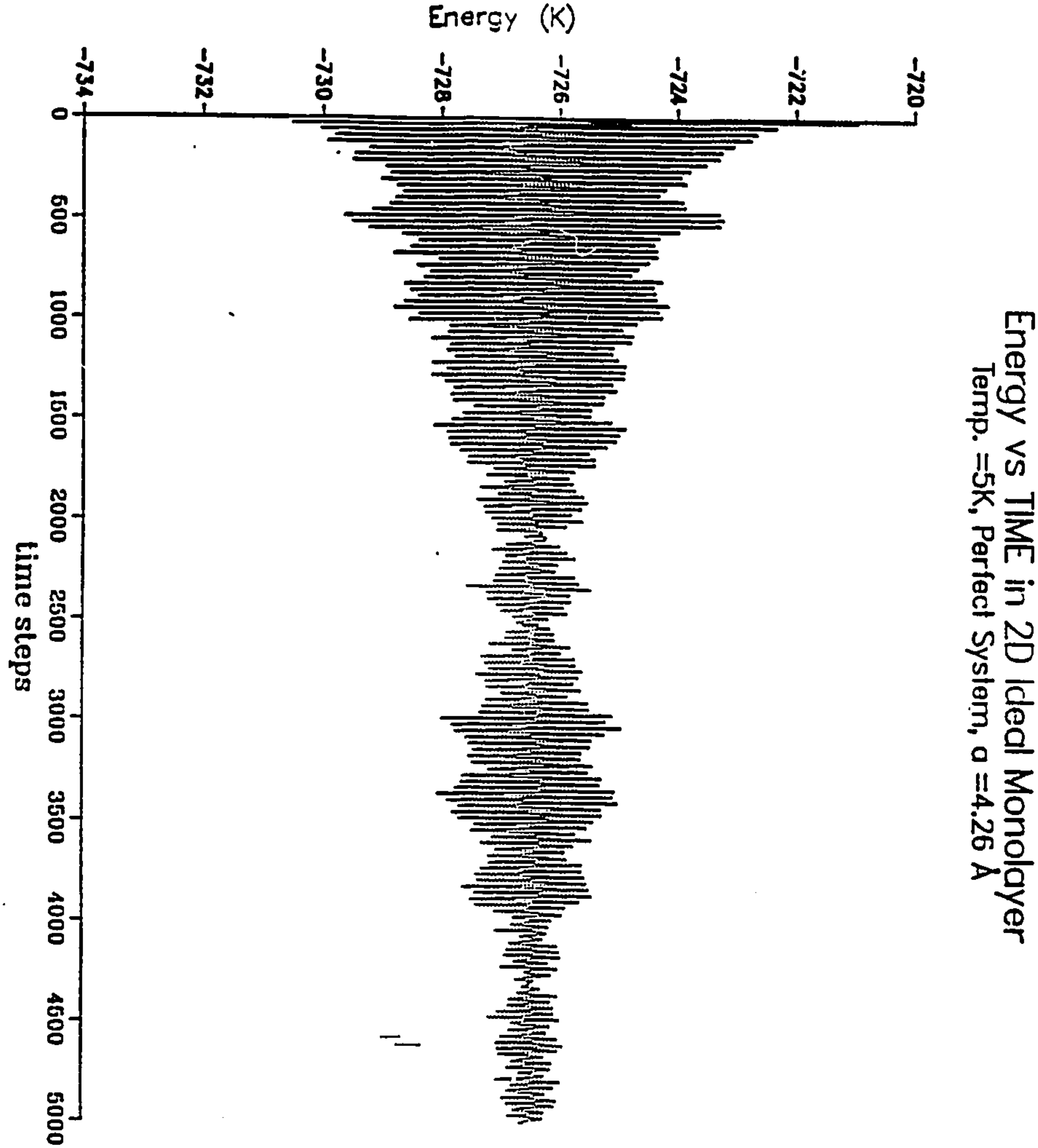

Figure 4.25: Evolution of the energy during NPT molecular dynamics without changing the value of M, in 2D at T=5.0K and $\Delta t = 2\ 10^{-14}$ seconds.

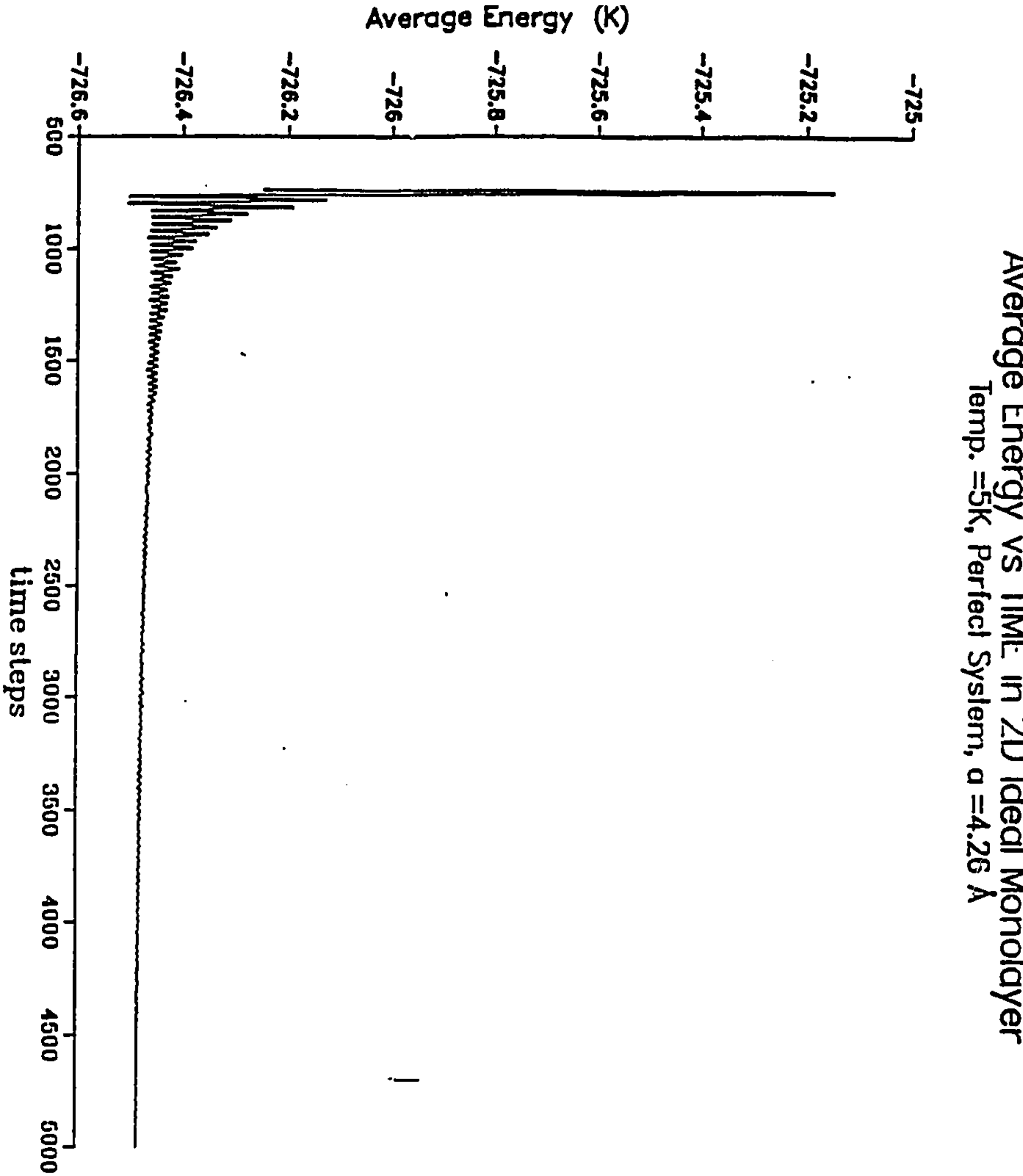

Figure 4.26: Evolution of the average energy during NPT molecular dynamics without changing the value of M, in 2D at T=5.0K and $\Delta t = 2\,10^{-14}$ seconds.

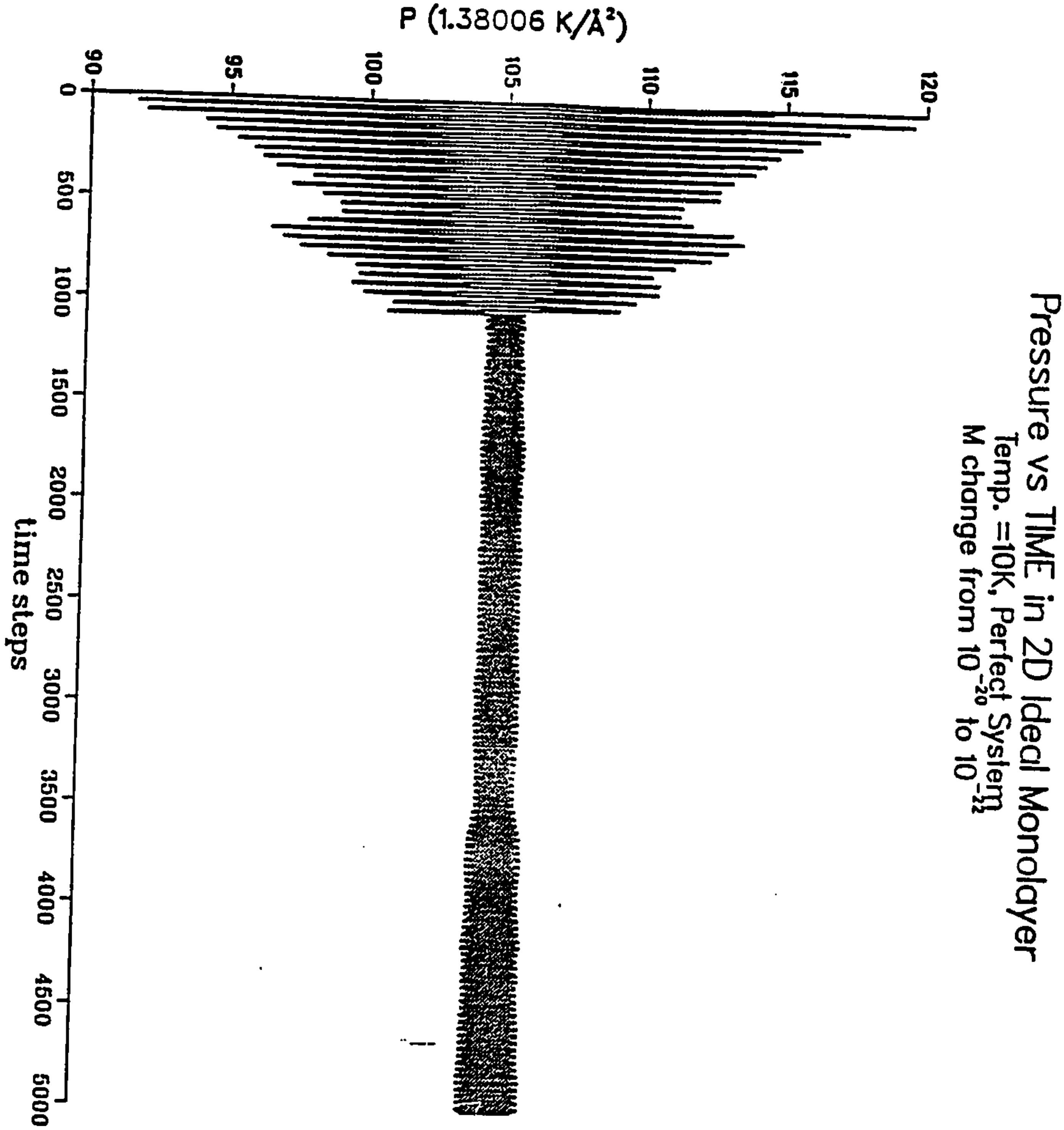

Figure 4.27: Evolution of the pressure during NPT molecular dynamics changing the value of M, in 2D at T= 10K and $\Delta t = 2\,10^{-14}$ seconds.

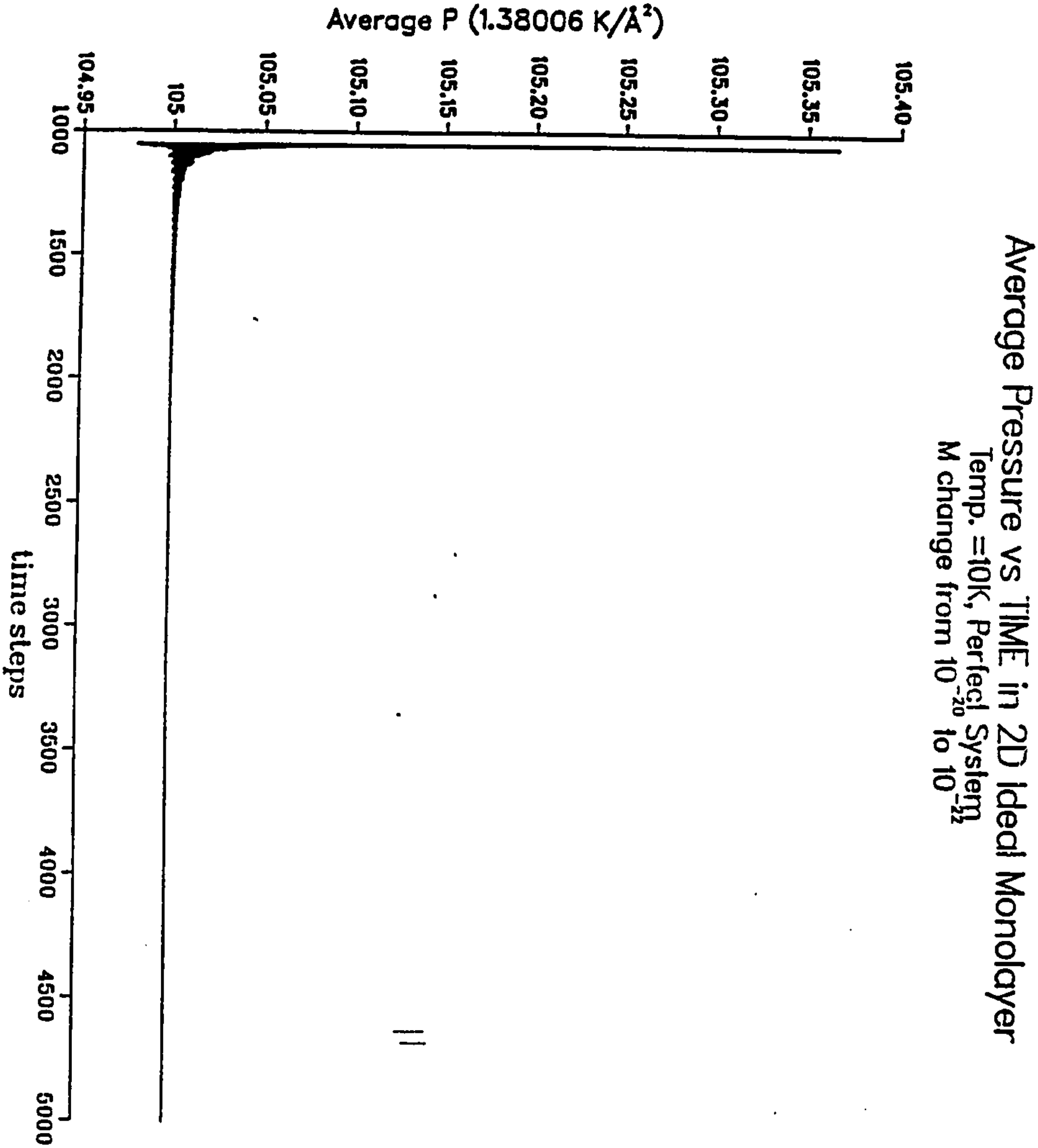

Figure 4.28: Evolution of the average pressure during NPT molecular dynamics changing the value of M, in 2D at T= 10K and $\Delta t = 2\,10^{-14}$ seconds.

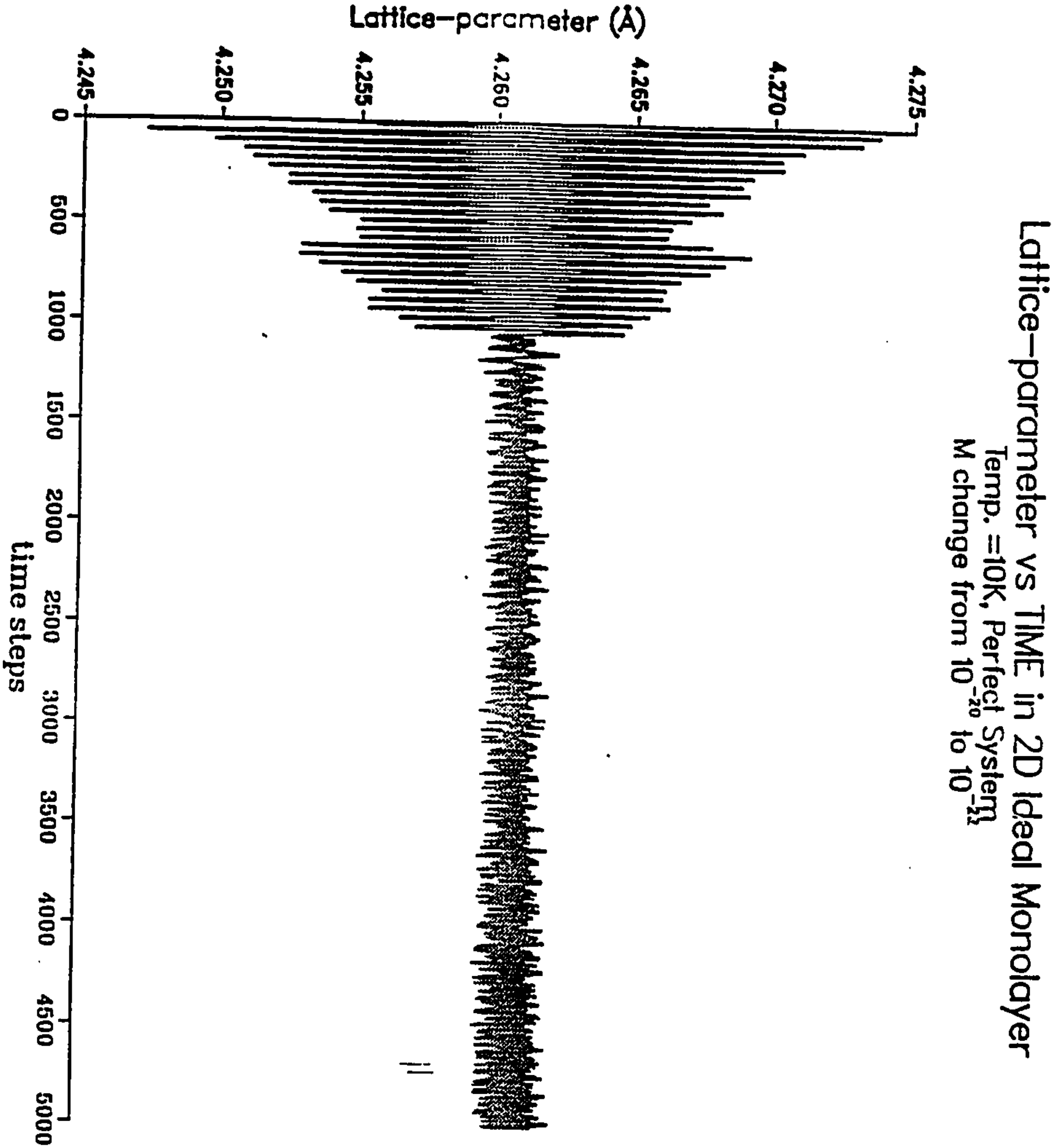

Figure 4.29: Evolution of the lattice parameter during NPT molecular dynamics changing the value of M, in 2D at T= 10K and $\Delta t = 2\ 10^{-14}$ seconds.

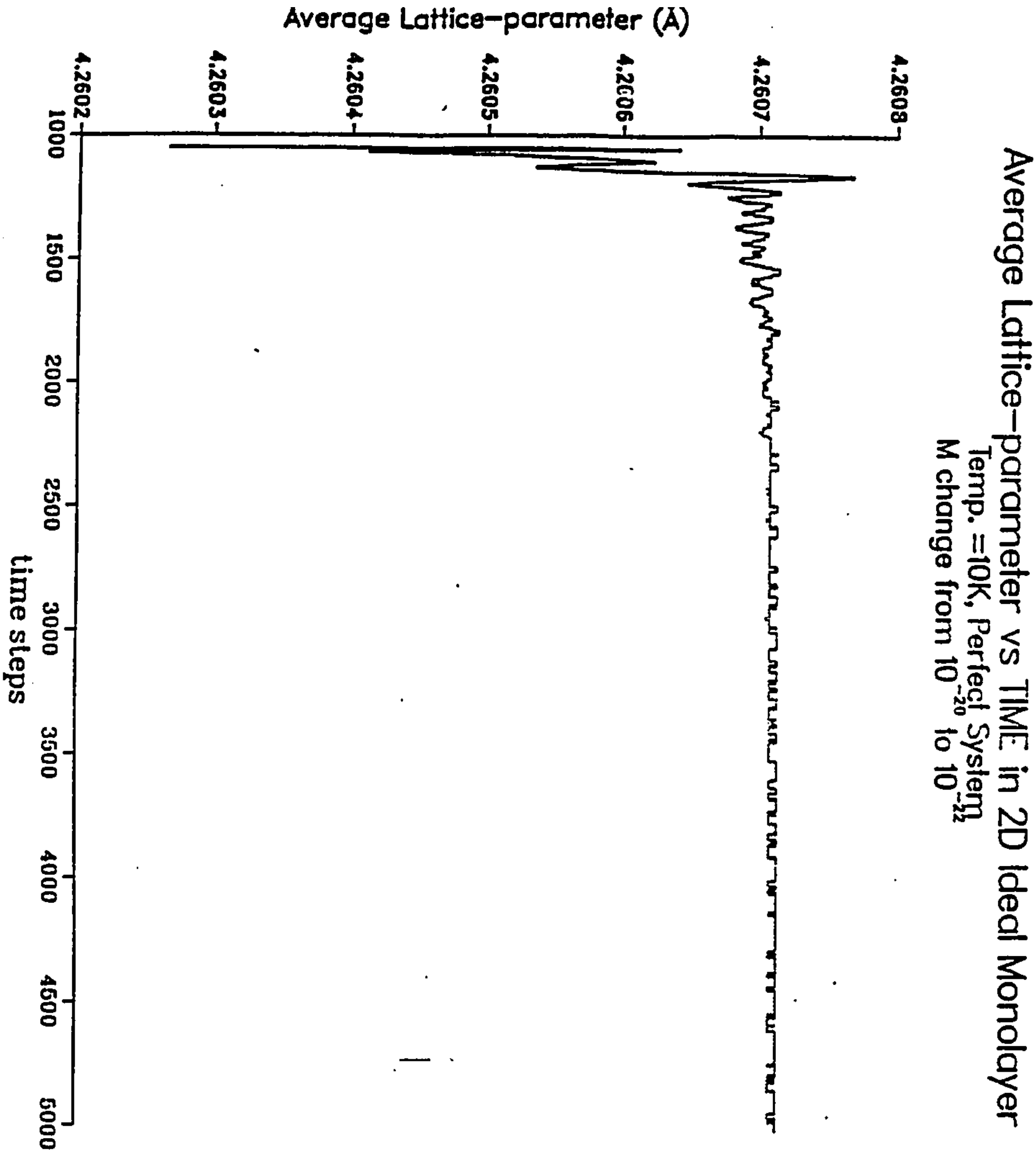

Figure 4.30: Evolution of the average parameter during NPT molecular dynamics changing the value of M, in 2D at T= 10K and $\Delta t =$ [illegible] 10^{-14} seconds.

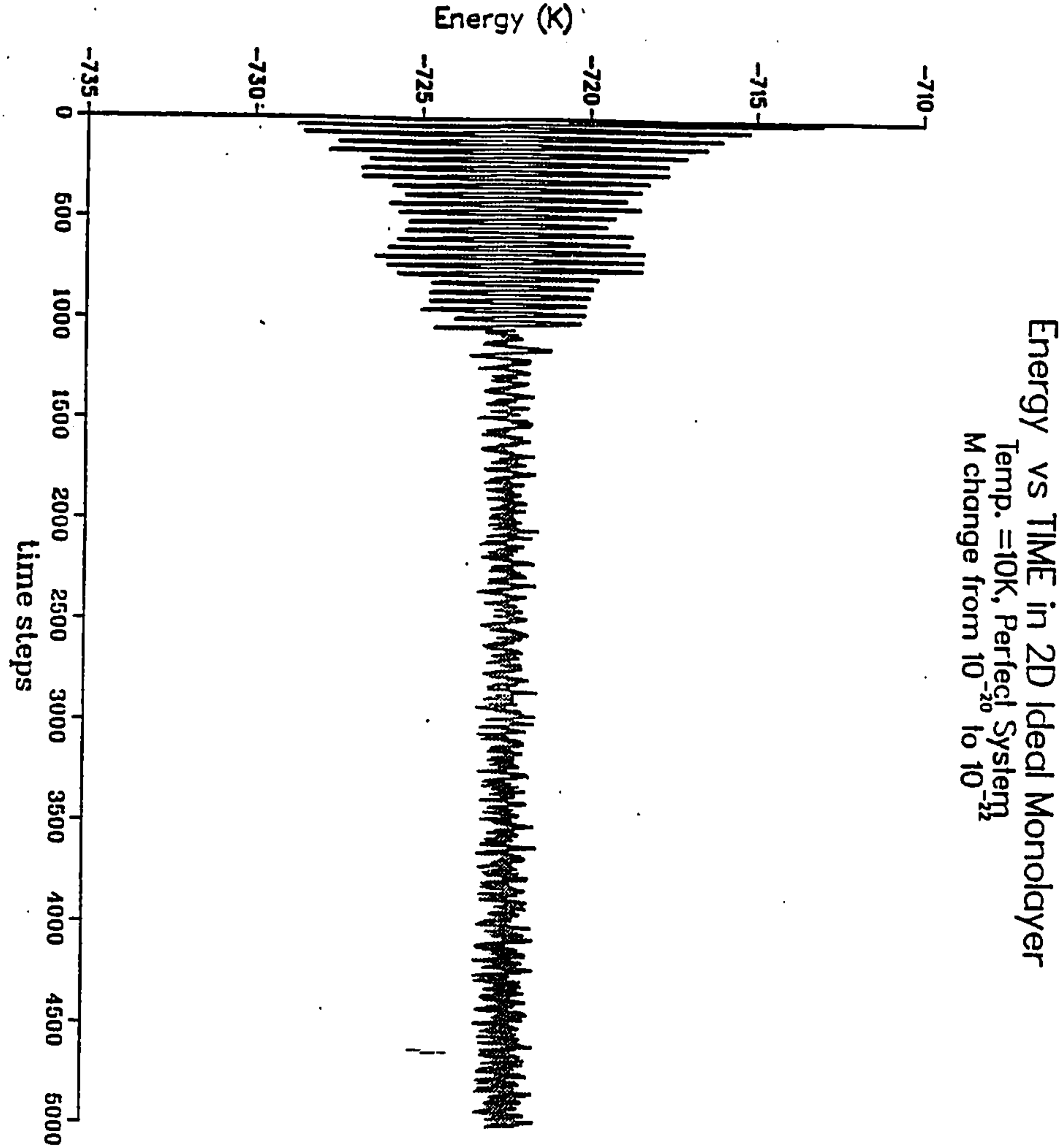

Figure 4.31: Evolution of the energy during NPT molecular dynamics changing the value of M, in 2D at T= 10K and $\Delta t = 2\,10^{-14}$ seconds.

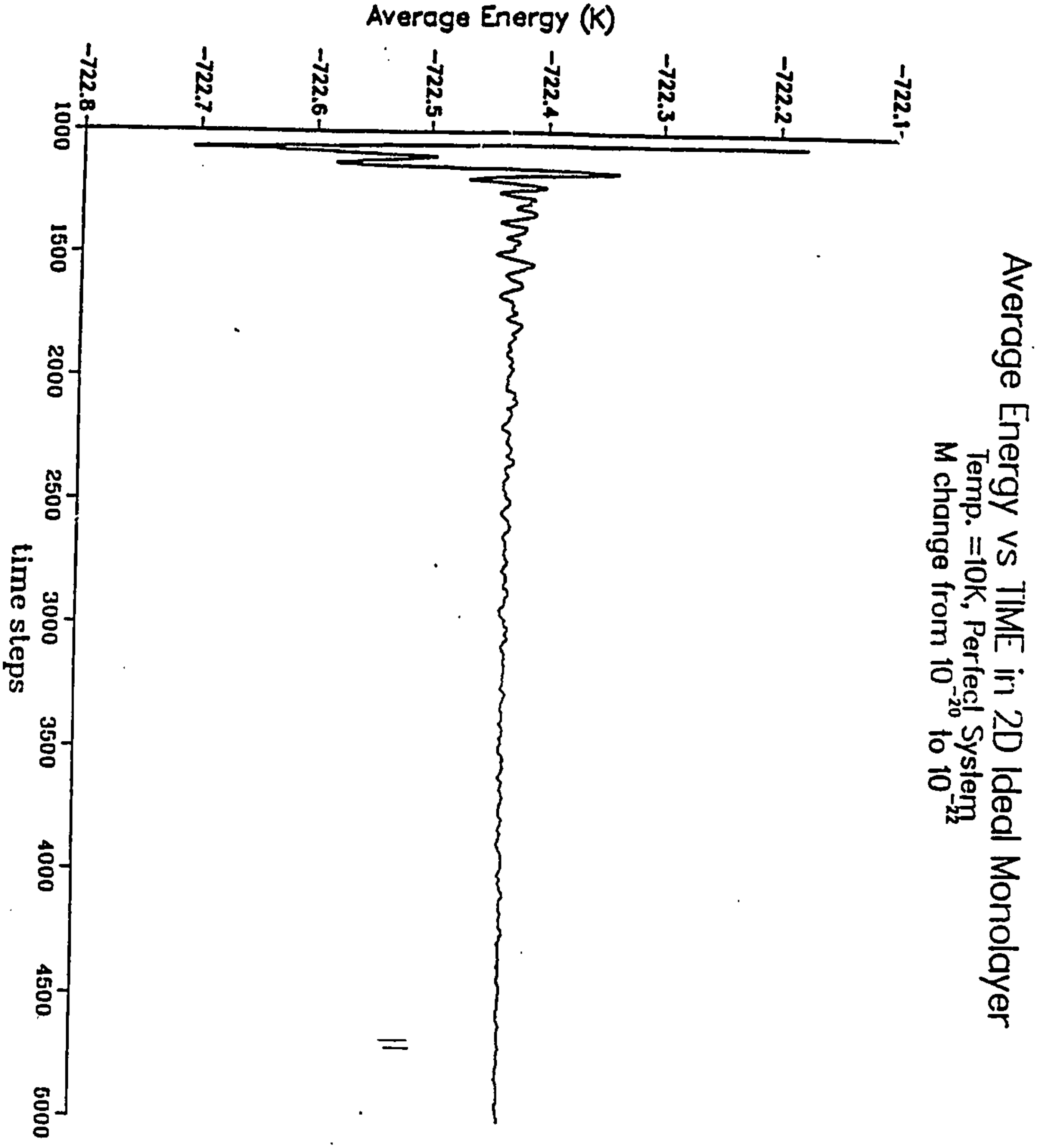

Figure 4.32: Evolution of the average energy during NPT molecular dynamics changing the value of M, in 2D at T= 10K and $\Delta t = 2\,10^{-14}$ seconds.

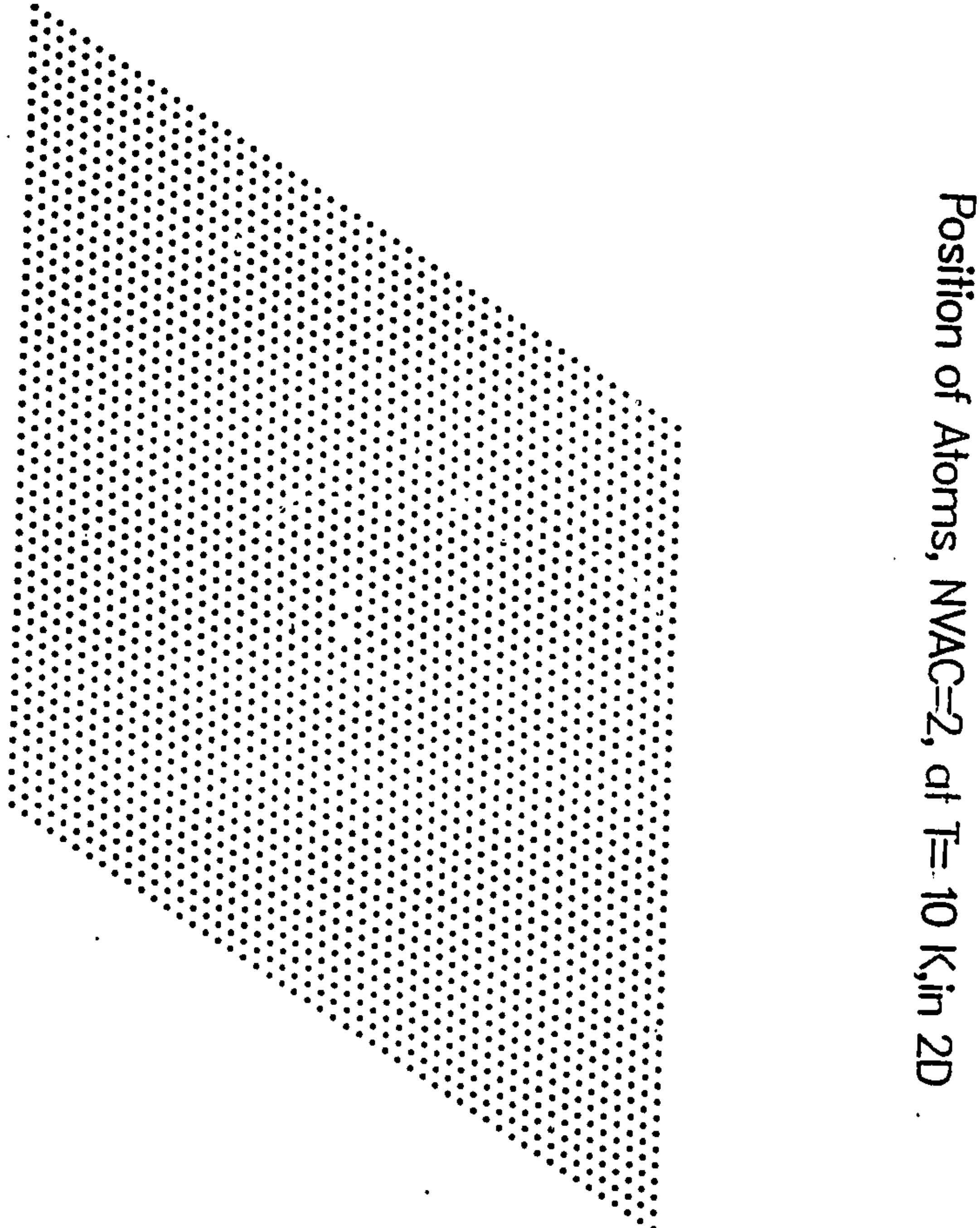

Figure 4.33: The position of atoms in final configuration for the interaction between two vacancies with a separation of 2a (a: Lattice-parameter) at t=5K, and a=4.26Å

Chapter 5

ENERGY CALCULATIONS:

5.1 Method of Energy Calculations:

The calculations are first performed at constant area, neglecting the activation area of the defects. In these circumstances a constant-area defect energy U_A can be written as;

$$U_A = \frac{1}{2} \sum_k \sum_l (U(r_{kl}) - U(r^o_{kl})) \tag{5.1}$$

Where,

r_{kl}: Distance between atoms i and j in a dislocated system.

r^o_{kl}: Distance between atoms i and j in a perfect system.

For constant pressure a correction to the constant-area energy U_A is necessary, the assembly relaxed at constant area will have an internal pressure or pressure of our system,

$$P = -\frac{\partial U(\mathbf{r})}{\partial \mathcal{A}} \tag{5.2}$$

which is not, in general, equal to the initial external pressure P_E. A Taylor expansion with respect to area change yields;

$$\Delta U = -(P - P_E)\,\Delta\mathcal{A} + \frac{K}{2\,\mathcal{A}}\,(\Delta\mathcal{A})^2 \tag{5.3}$$

Where K is an "area" modulus, and ΔU is now the change in energy of the total system.

This expression is easily minimized with respect to $\Delta\mathcal{A}$ to determine the area change necessary to convert to constant pressure condition, yielding;

$$\Delta\mathcal{A} = \frac{\mathcal{A}}{K}\,(P - P_E) \tag{5.4}$$

It can be verified that the internal pressure after applying the correction given by the equation (5.4) equals the external pressure P_E. It is found that three, four or more iterative applications are necessary to achieve a constant pressure state see section 4.4 of chapter 4.

The correction (5.4) seems to have been suggested first by Hardy (1960) [39] for a three dimensional point deffects.

In constant pressure the pair energies are assummed to give an energy U_P.

Finally, the external pressure term must be incorporated to give a true defect internal energy;

$$\Delta U = U_P + P_E\,\Delta\mathcal{A} \tag{5.5}$$

In our calculations U_P represents the energy difference between the perfect and dislocated systems at the same pressure, and $P_E\Delta\mathcal{A}$ is the work done to put the vacancy in the system, which represents the work required to change the area to return the system to pressure P.

5.2 Vacancy Energy:

The vacancy formation energy, E_v^f, is the difference between the potential energy E_d, computed when the vacancy is present in the computational cell, and the potential energy E_p, computed when the computational cell is perfect. In the finite system vacancy formation sequence, the number of atoms is conserved in the calculations.

$$E_v^f = \left[\frac{E_d}{N - n_{vac}} - \frac{E_p}{N}\right] (N - n_{vac}) \tag{5.6}$$

Where n_{vac} is the number of vacancies and N the total number of atoms.
E_v^f will represent U_p in the equation (5.5). The atom which was extracted to give a vacant site at the center of the cell was replaced at the system surface of the dislocated cell.

$$P_E \, \Delta\mathcal{A} = P_E \left[\left(\mathcal{A}_d + \frac{n_{vac}}{N}\,\mathcal{A}_p\right) - \mathcal{A}_p\right] \tag{5.7}$$

Where $\mathcal{A}_d$ is the area of the cell when the vacancy is present and $\mathcal{A}_p$ is the area of the perfect cell.

The total vacancy energy is be given by:

$$E_{vac} = E_v^f + P_E \, \Delta\mathcal{A} \tag{5.8}$$

in vacancy energy calculations n_{vac} is equal to one.
E_d and E_p are given for the same pressure.

In case of two vacancies n_{vac} is equal to two.

5.3 Vacancy Migration or Activation Energy:

5.3.1 Introduction:

The work on migration energy calculations was introduced by Johnson [5] who also led the discussion on migration energy calculation techniques.
Johnson restricted his discussion to migration energy calculations for vacancies and interstitials, predominantly in the fcc crystal structure; he included ample reference to the bcc structure to give a complete exposition. Simulation of the positional interchange between a vacancy and one of its nearest-neighbor atoms provides an example in which the computational technique is staightforward.
Seveval dynamical methods simulation of defect migration were treated [40] [41].
Wilson and Bisson [42] present the dynamical method simulation of helium migration in bcc iron.
Guinan [43] performed a fully dynamical simulation of self interstitial diffusion in tungsten(m3). Same result was found by Johnson [5] for bcc iron(m).
In our case we present the dynamical method simulation of xenon migration in a triangular lattice of xenon atoms, two cases are treated: first the 2D ideal Xe monolayer , second the 3D adsorbed Xe monolayer on silver(111).

5.3.2 Energy Calculations:

In triangular lattice, the activated atom or vacancy migration consists of a positional exchange between an atom A at site P_2 and a vacancy at site P_1, which is a first neighbor of site P_2 as shown in figure (5.1) .

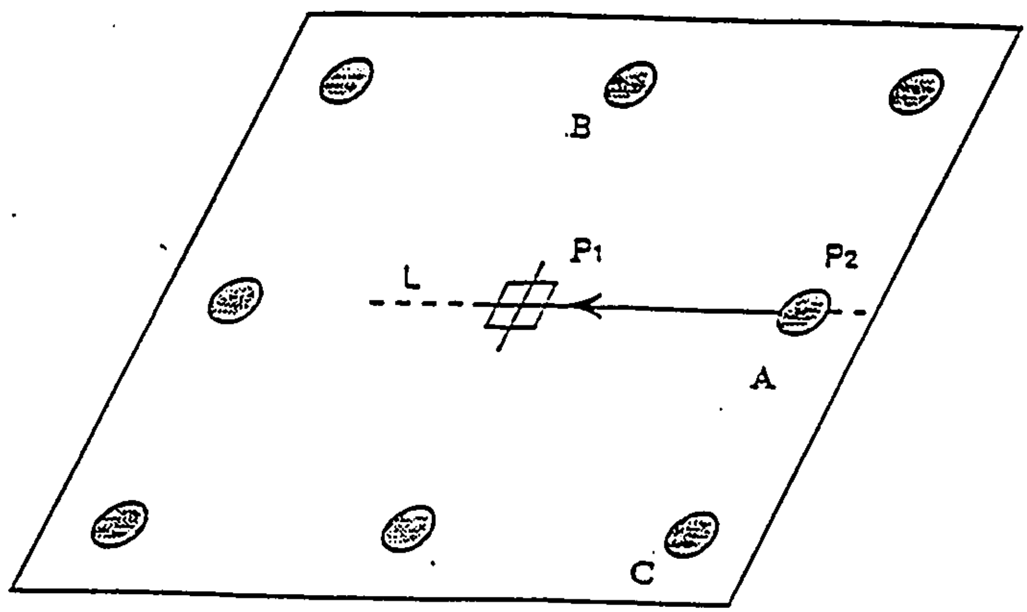

Figure 5.1: The transition motion corresponds to atom A moving along the line L from P_2 to P_1.

Simulation of the activated atom consists of finding the transition from the site P_2 to the site P_1. The activation energy for the transition is:

$$E^t_{Act}(i) = E_A(i) - E_d \tag{5.9}$$

Where i indicates the transition between the site P_2 and site P_1, E_A is the potential energy computed when the atom at site P_2 is activated and E_d is the potential energy computed when the vacancy is present. The number of atoms is conserved. In this case the total activation energy will be given by:

$$E_{Act}(i) = E^t_{Act}(i) + P_E\ [\mathcal{A}_{Act}(i) - \mathcal{A}_d] \tag{5.10}$$

Where $\mathcal{A}_{Act}$ is the area of the cell when the atom at site P_2 is activated and $\mathcal{A}_d$ same as in equation(5.7).
The energies E_A and E_d are given for the same pressure.

The configuration energy $E_{Act}(i)$ is obtained by holding atom A fixed at position $\mathbf{r}_A + \mathbf{i}a$, where $\mathbf{i}$ represent deplacement vector given to the atom A (a: lattice parameter) and allowing all other atoms to relax to statistic equilibrium positions.

5.4 Errors:

Our results contain errors due to the method of calculation and it is different for each simulation method used.

In NPT molecular dynamics simulation, we cannot fix the lattice-parameter at a constant value, the parameter fluctuates in time see figure 8.19. For a small error $\pm 10^{-3}\mathring{A}$ in lattice-parameter, the error found for the energies is about $\pm 80K$, which is due to the thermal fluctuation effects.

In case of NVT molecular dynamics simulation, the lattice parameter is kept constant, but to get the constant pressure we used the iteration method which is one reason for the errors. For an error of $\Delta P = \pm 10^{-3}(10^{+16}\ k_B\ K/\mathring{A}^2)$, the error found for the energy is about $\pm 28K$, in two dimensions. In three dimensions the energy error is about $\pm 10K$.

The other errors are related to average calculations of our quantities as pressure, lattice-parameter and energy, but they are not comparable to the errors described before.

Chapter 6

ACTIVATION ENERGY RESULTS AND DISCUSSION

6.1 Vacancy Energy:

6.1.1 Vacancy Energy Calculations:

All our energies are given in units of K and the energies used for the calculations are average values after a number of time steps. Our simulations were performed using a tringular lattice of 51 atoms each side 2601 atoms in total. The vacancy energies are obtained by the difference between the perfect and dislocated systems at same pressure, plus the work done to put the vacancy in the system. By NVT molecular dynamics method the constant pressure is obtained by three or four iterations see chapter 4.4. However, by NPT molecular dynamics method we insure that the pressure is kept constant which is

Table 6.1: Vacancy energy given by different simulation methods.

Method of Simulation	Lattice-parameter (Å)	T (K)	Vacancy Energy (K)
NVT Molecular Dynamics Method	4.26	5 K	1821
NPT Molecular Dynamics Method	4.26	5 K	1884
B. Joós and M.S. Duesbery Result	4.26	0	1770

provided by the algorithm itself.

Energies of vacancies in Xe have been calculated by different simulation methods, the results are given in table 6.1 for two dimensional Xe monolayer. The difference in energies of vacancies is due to the method used and its precision. Joós and Duesbery [36], results are for zero temperature.

The other two are done at 5K. Temperature fluctuations introduce errors in both the NVT and NPT results. The NVT is however more precise than the NPT because there are no fluctuations in the lattice constant in the NVT case. In three dimensional adsorbed xenon on silver(111) the vacancy energy at temperature 5 K is given in table 6.2 .

Table 6.2: Vacancy energy in 3D adsorbed Xe on Ag(111).

Method of Simulation	Lattice-parameter (Å)	T (K)	Vacancy Energy (K)
NVT Molecular Dynamics Method	4.26	5 K	1849

6.1.2 Temperature Dependence:

The Debye temperature θ_D is equal to $66K$ [44] for the xenon monolayer in two dimensions.

The vacancy energy is expected to rise linearly with temperature for temperatures below θ_D . But as for the calculation by Grossmann [37], we find a very small increase whose rate is nearly masked by fluctuations. See figure 6.2.

The increase in energy is due to two effects, first to the actual thermal vibrations, second to the area change to maintain constant pressure see figure 6.11 and 6.12.

From the equipartition theorem it follows that in the elastic regime the kinetic energy and the potential energy vary linearly with the temperature. It is the difference in the energy of the perfect and defected lattice which is important hence at these temperatures there will be little change in the potential energy of the defect. However to maintain constant pressure an area change is done. The related work contributes the largest part in the change of energy of the defect.

6.2 Activation Energy:

6.2.1 Activation Energy Calculations:

The activation energy calculations have been done using a NVT molecular dynamics simulations, which directly simulates the dynamical behavior of Xe atoms in a triangular system at finite temperature. The defect movement arises naturally from fluctuations in the atom positions as a consequence of the thermal vibrations. This movement constitutes a transition from an initial equilibrium configuration where the vacancy is at site P_1 see figure 5.1, to a final equilibrium state where the vacancy is at site P_2. According to the absolute rate theory model, this transition can take place if the system can achieve an intermediate unstable equilibrium configuration state called the saddle point configuration from which the final equilibrium state is possible.

The simulation of vacancy migration consists of finding the energy of configurations at different points between the site P_1 and the site P_2. These results are indicated in table 6.3 for two dimensions and in table 6.4 for three dimensions. The plot of $\{ E_{Act}(i) \}$ versus displacement $\{ \mathbf{r}_A + \mathbf{i}\, a \}$ on the L axis see figure 5.1 (a is the lattice parameter and i the vector displacement). The plot is given by figure 6.3 for the 2D Xe ideal monolayer, and by figure 6.4 for the 3D adsorbed Xe monolayer on the (111) face of silver.

DX and DY are the displacements of the activated atom, $\langle Z \rangle$ represents the average Z coordinate of the activated atom. At the initial equilibrium configuration the vacancy is at $DX = 0$ and $DY = 0$ in the triangular lattice.

The figures 6.3 and 6.4 indicate the maximum energy which represents the saddle point energy E_s for the activation energy. This saddle point occurs when the activated atom is at midpoint between sites P_1 and P_2, which is typical of

Table 6.3: Activation energy function of the displacement DR of the activated atom in units of a (a:Lattice-parameter) from the site P_2 to the site P_1, at T= 5K and at constant pressure $P = 73.86K/\AA^2$, in 2D Xe ideal monolayer.

Displacement DR	DX(units of a)	DY(units of a)	Activation Energy (K)
0.1	-0.1	0	71.4358
0.2	-0.2	0	307.9401
0.3	-0.3	0	597.4698
0.4	-0.4	0	834.2668
0.5	-0.5	0	921.2230
0.6	-0.6	0	834.2668
0.7	-0.7	0	597.4698
0.8	-0.8	0	307.9401
0.9	-0.9	0	71.4358
1.0	0	0	1.5583

Table 6.4: Activation energy function of the displacement of the activated atom from the site P_2 to the site P_1, at T= 5K and at constant pressure $P = 72.34K/\AA^2$, in 3D Xe adsorbed monolayer on silver(111).

DX(unit of a)	DY(unit of a)	$\langle Z\rangle(\AA)$	Activation Energy (K)
-1.0	0	3.528	0.0
-0.1	0	3.477	61.4094
-0.3	0	3.336	548.1665
-0.5	0	3.299	768.9738
-0.7	0	3.336	548.1665
-0.9	0	3.477	61.4094
0	0	3.528	1.3455

the activation energy in a first-neighbor interaction model.

The saddle point is a single maximum. There are cases where the plot of { $E_{Act}(i)$ } versus displacement { $\mathbf{r}_A + \mathbf{i}\, a$ } of the activated atom exhibits two maximums which can be symmetrically or not located about the midpoint.

Beeler [41] found an asymmetry for two maxima about the midpoint in alloys.

The figure 6.5 gives a comparison of the activation energy configuration between two dimensions and three dimensions. The values of the activation energy in three dimensions are smaller compared to those given in two dimensions. The explanation to this result is due to the trajectory taken by the activated atom, in the three dimensional case when the activated atom goes from the site P_2 to the site P_1 see figure 5.1, it follows the trajectory given by figure 6.6 in the plane L perpendicular to the system where the coordinates X and Y of the activated atom are kept fixed, and its Z coordinate fluctuate as shown in figure 4.17 and 4.18. As we can see in figure 6.6 the activated atom Z coordinate gets smaller when we get clouser to the midpoint, while the Z coordinates of neighboring atoms 1302 and 1351 with a separation of $\frac{\sqrt{3}}{2}\, a$ (a is the lattice parameter) in figure 6.1 increase, this increase is the largest and is about 1.17Å from the initial state at the midpoint see table 6.5.

For the second-neighbors with a separation of $\frac{\sqrt{7}}{2}\, a$ the average of Z coordinate decrease about 0.11Å . The third-neighbors with a separation of $\frac{3}{2}\, a$ the average of Z coordinate increase about 0.04Å from the initial state which represent about 3.53Å . The other atoms kept an average value of 3.53Å.

This show that the activation of the atom 1352 from the site P_2 to the midpoint between the sites P_2 and P_1 changes the value of the average of Z coordinate for the three first nearest-neighbor.

The trajectory followed by the activated atom keeps it at a larger distance from its neighbors than in two dimensions. It is interesting to note that the activated atom gets closer to the surface while its neighbors rise. It takes less energy to increase the Z coordinate by a given amount than to decrease it. The neighbors being more numerous set the tone.

6.2.2 Temperature dependence of the activation energy:

Activation energy:

The activation energy increases by about 10 percent in the temperature range 5 to 65K. In the two dimensional case the activation energy increases from the value of 921K to 993K in the range 5 to 65 K which represents a 7.8 percent increase. For the three dimensional system the activation energy increases from the value of 768K to 851K in the temperature range 5 to 50 K which represents a 10.8 percent increase.
This is an average activation energy. In the thermodynamic sense it is an average internal energy of activation not a free energy of activation. When the temperature is raised, the instantaneous activation energy fluctuates in time with larger and larger amplitudes, the minima getting deeper and the maxima higher.
To get an activation energy which would reflect the increased probability of hopping from one site to the next, the entropic contribution should be calculated. This negative contribution to the free energy increases with the fluctuations.

Comparison with other energies:

The rates of change of some relevant energies are given in table 6.6 together with those of the activation energies in two and three dimensions.

Table 6.5: Average Z coordinate for some nearest-neighbor of the activated atom for perfect system, dislocated system, with a displacement of the activated atom of $0.3a$ and with a displacement of $0.5a$ from its initial position (the displacement is done along X axis), see figure 6.1.

Number of atom	P. System	Disl. System	Displ.=0.3a	Displ.=0.5a
1300	3.5355	3.5380	3.4288	3.4318
1302	3.5437	3.5318	4.3348	4.7108
1351	3.5283	3.5312	4.2988	4.6971
1352	3.5331	3.5282	3.3365	3.2998
1353	3.5453	3.5278	3.4441	3.4236
1402	3.5420	3.5387	3.4418	3.4339
1403	3.5273	3.5518	3.5579	3.5736
1404	3.5323	3.5270	3.5323	3.5350

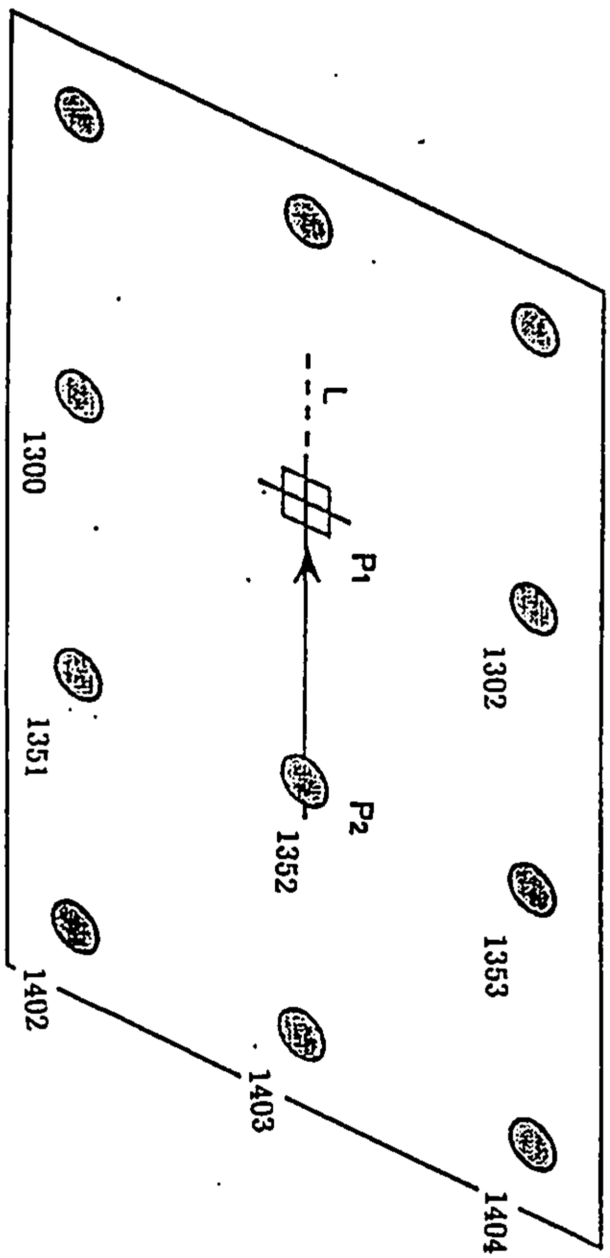

Figure 6.1: Positions of the activated atom nearest-neighbor

Table 6.6: The slope of the variation of different energies with the temperature.

Different Energies	Slope
Energy per Atom in two dimensions	0.66
Energy of Vacancy in two dimensions	1.40
Activation Energy in two dimensions	1.32
Energy per Atom in three dimensions	1.5
Activation Energy in three dimensions	2.0

The rate of change of the different energies, energy by atom in perfect system, energy of vacancy and activation energy with the temperatute is given in table 6.6 from the figures 6.2, 6.7, 6.8, 6.9 and 6.10.

The first result is that the rate of change of these energies with temperature is larger in three-dimensions, the second result is that the rate of change of the activation energy with temperature is larger than the rate of change of the energy by atom with temperature in both the two dimensional and three dimensional cases.

In two dimensions the rate of change of the energy of the vacancy is larger and closer to the rate of change of the activation energy.

The lattice parameters used for the calculations of the energies at different temperature for a constant pressure are given in 6.11 and 6.12. These figures show that the lattice parameters increase more in the two dimensional case in the range 5 to 65 K, than in three dimensional case.

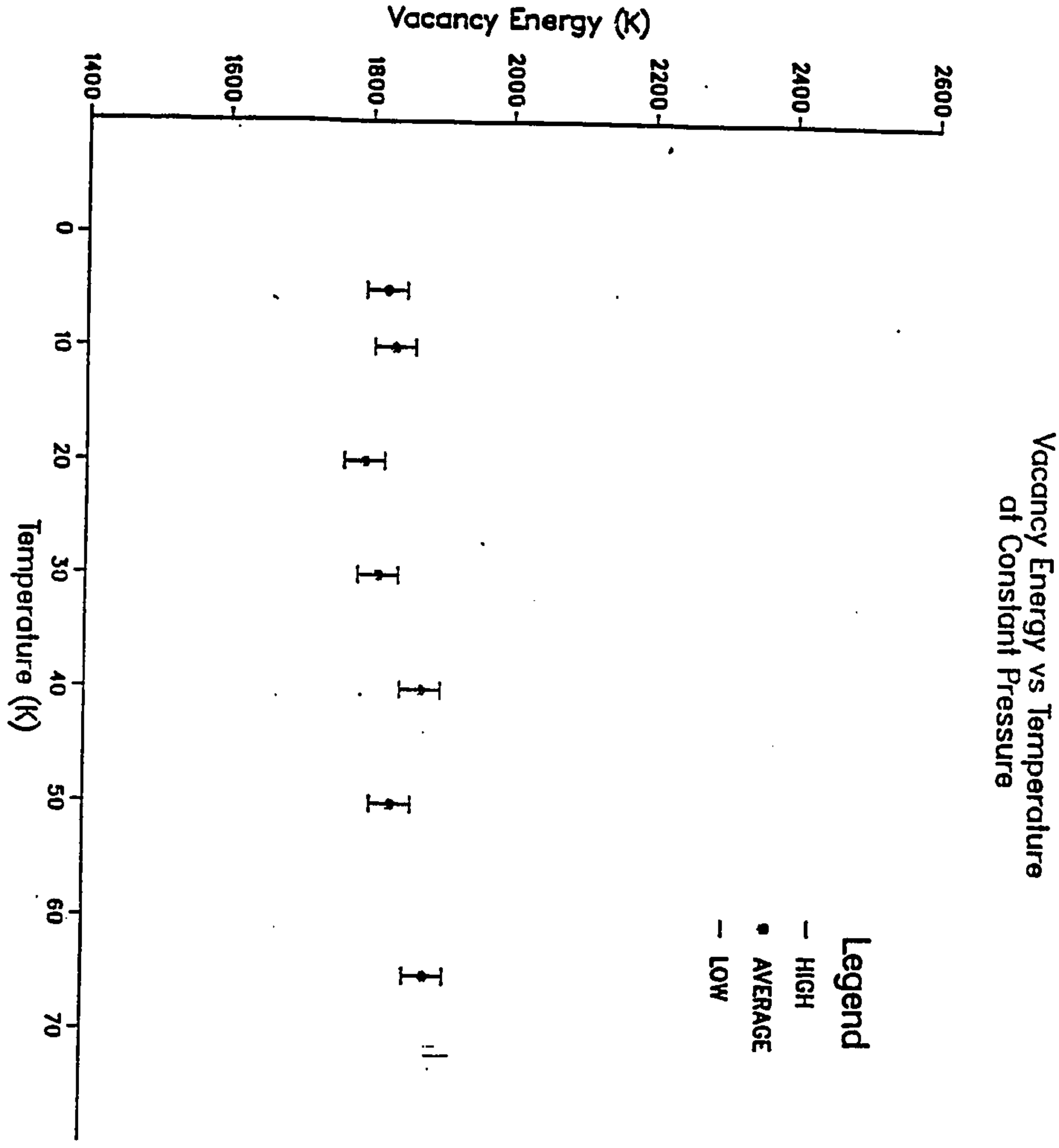

Figure 6.2: Variation of the vacancy energy as a function of Temperature at constant pressure $P = 73.86K/\AA^2$.

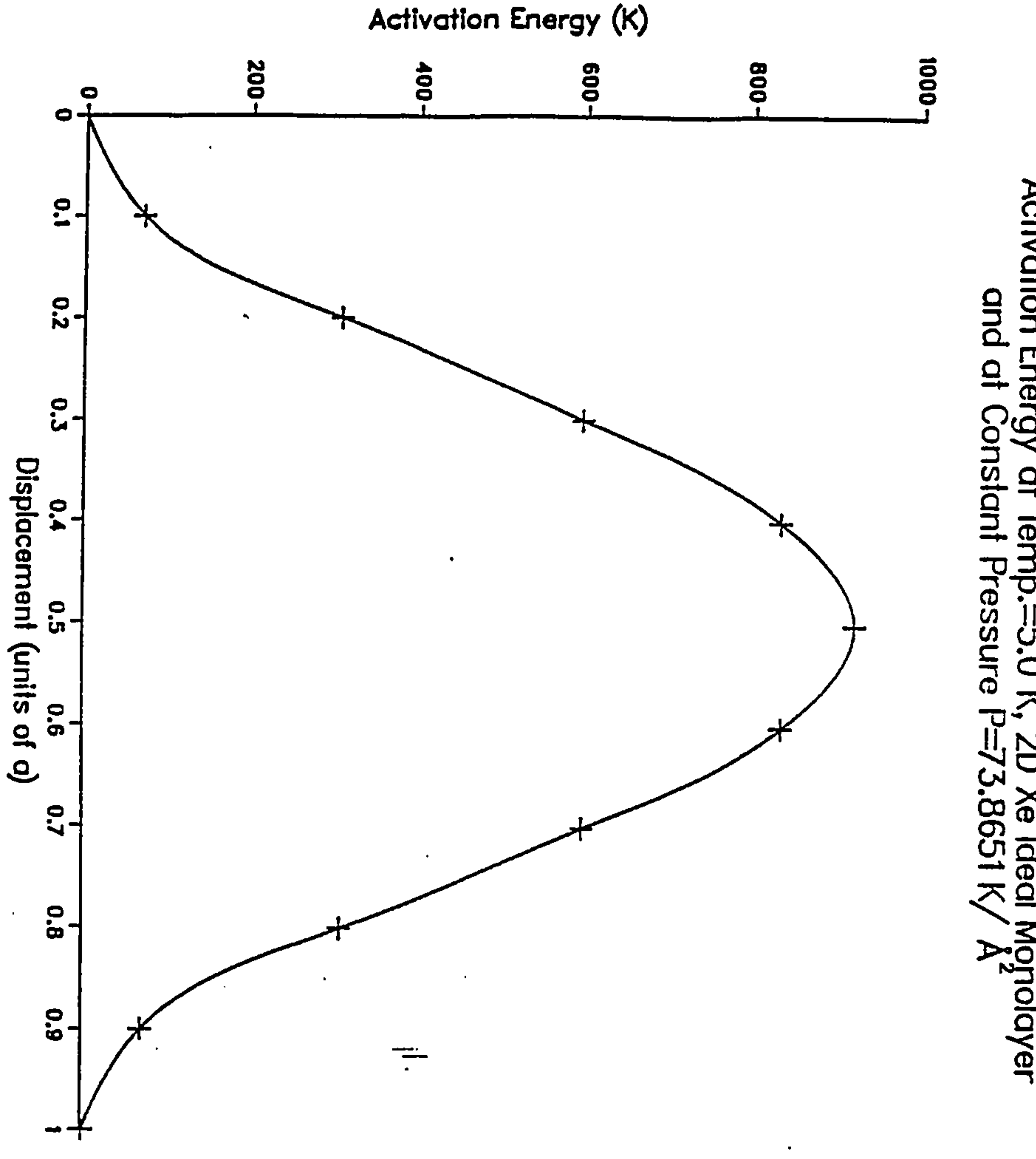

Figure 6.3: Activation energy transition from site P_2 to site P_1 in 2D Xe monolayer , at temp. = 5 K and at constant pressure $P = 73.86K/\mathring{A}^2$.

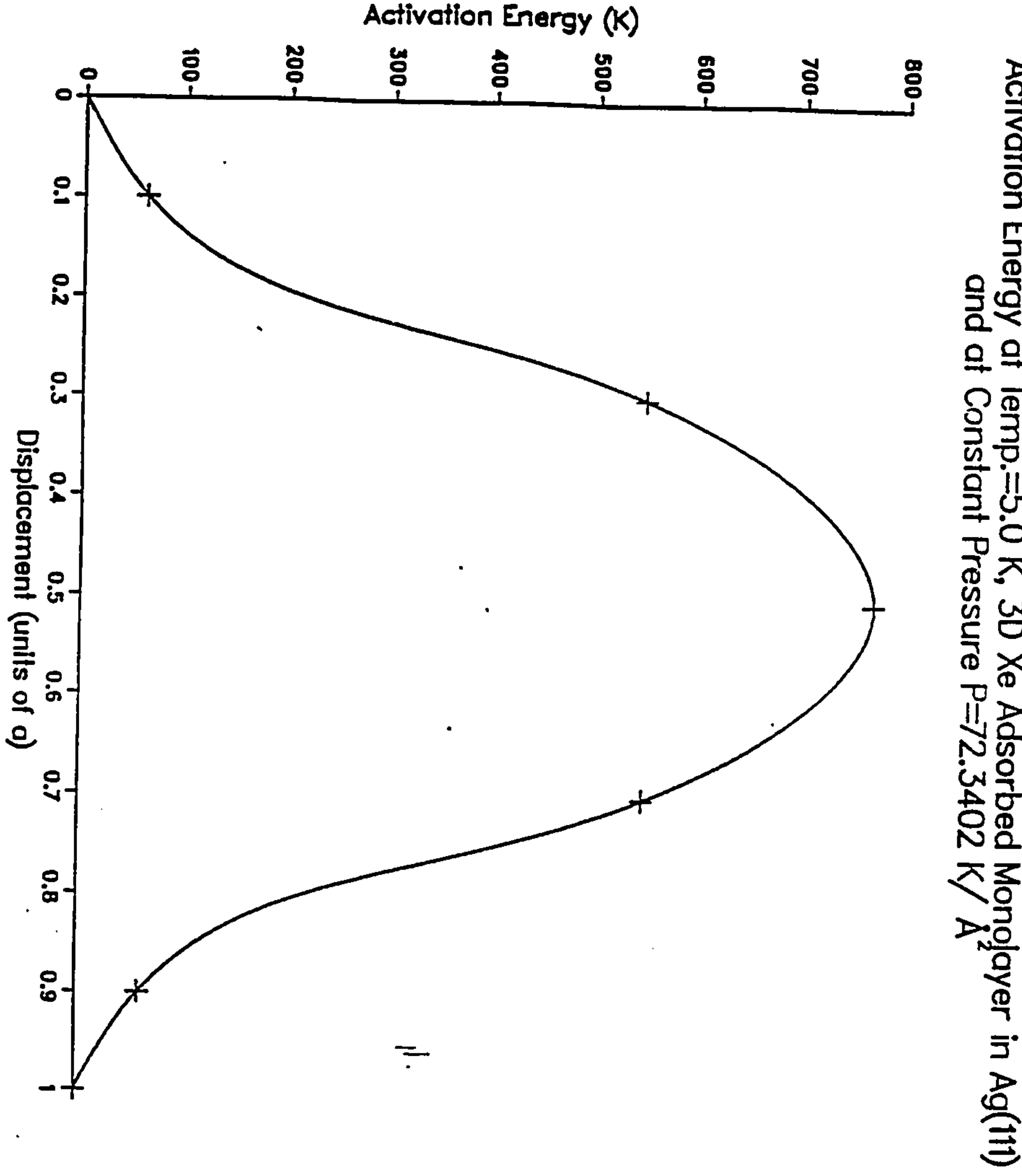

Figure 6.4: Activation energy transition from site P_2 to site P_1 in 3D Xe adsorbed monolayer on Ag(111), at temp. = 5 K and at constant pressure $P = 72.34K/\mathring{A}^2$.

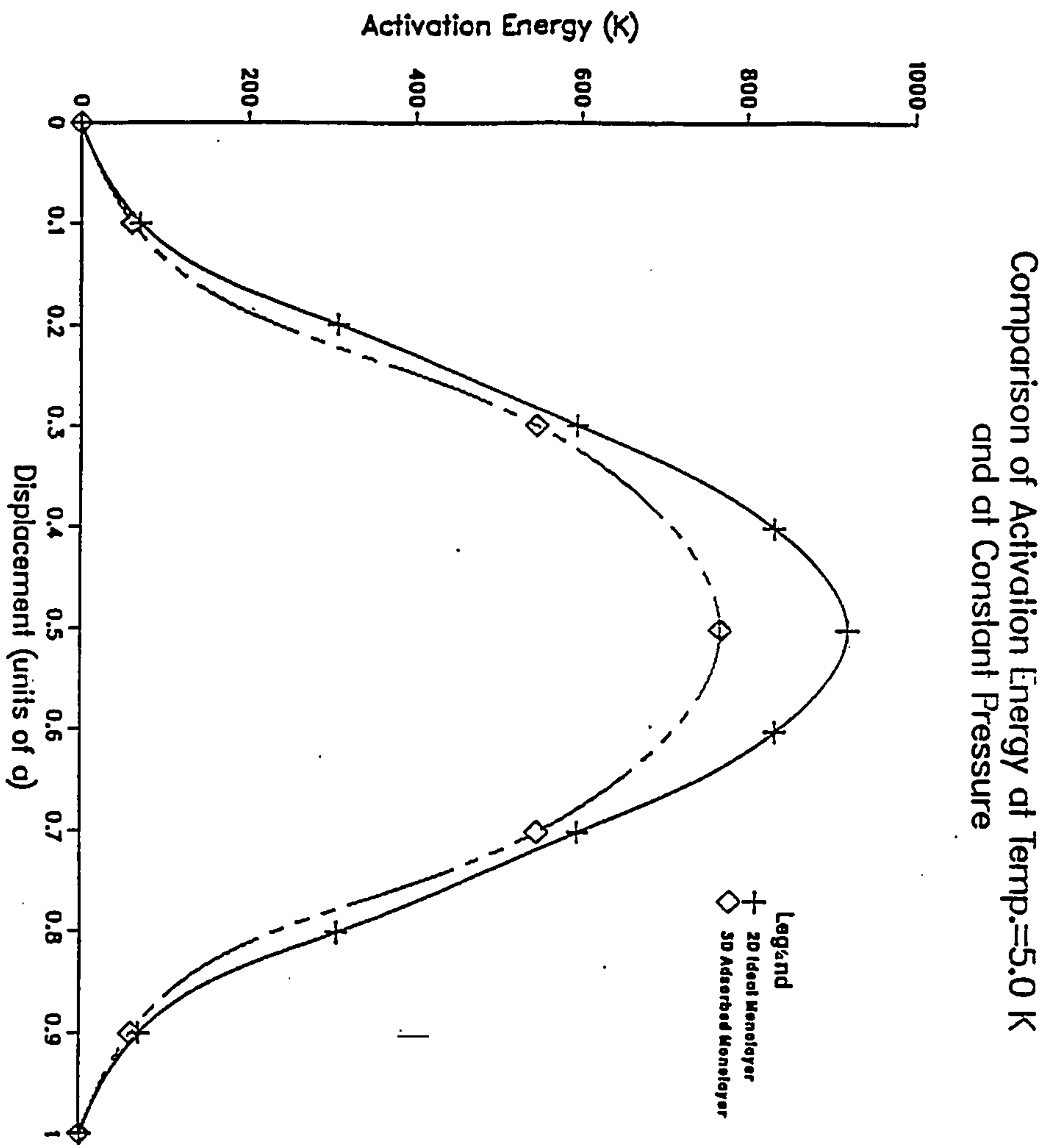

Figure 6.5: Comparison between the activation energy transitions in 2D and 3D at temperature of 5 K.

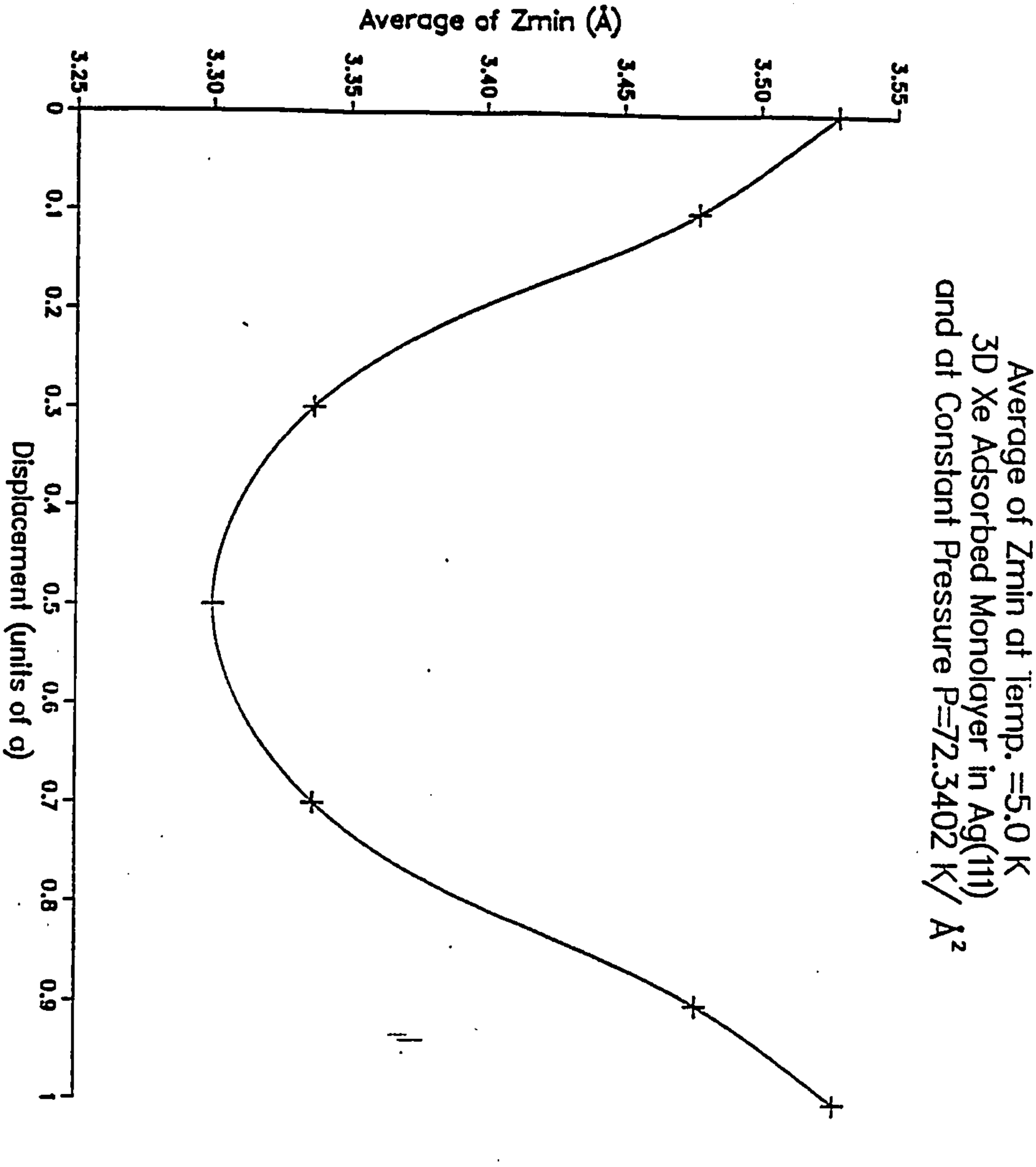

Figure 6.6: Trajectory of the average of Z coordinate of the activated atom from site P_2 to site P_1 at temp. = 5K.

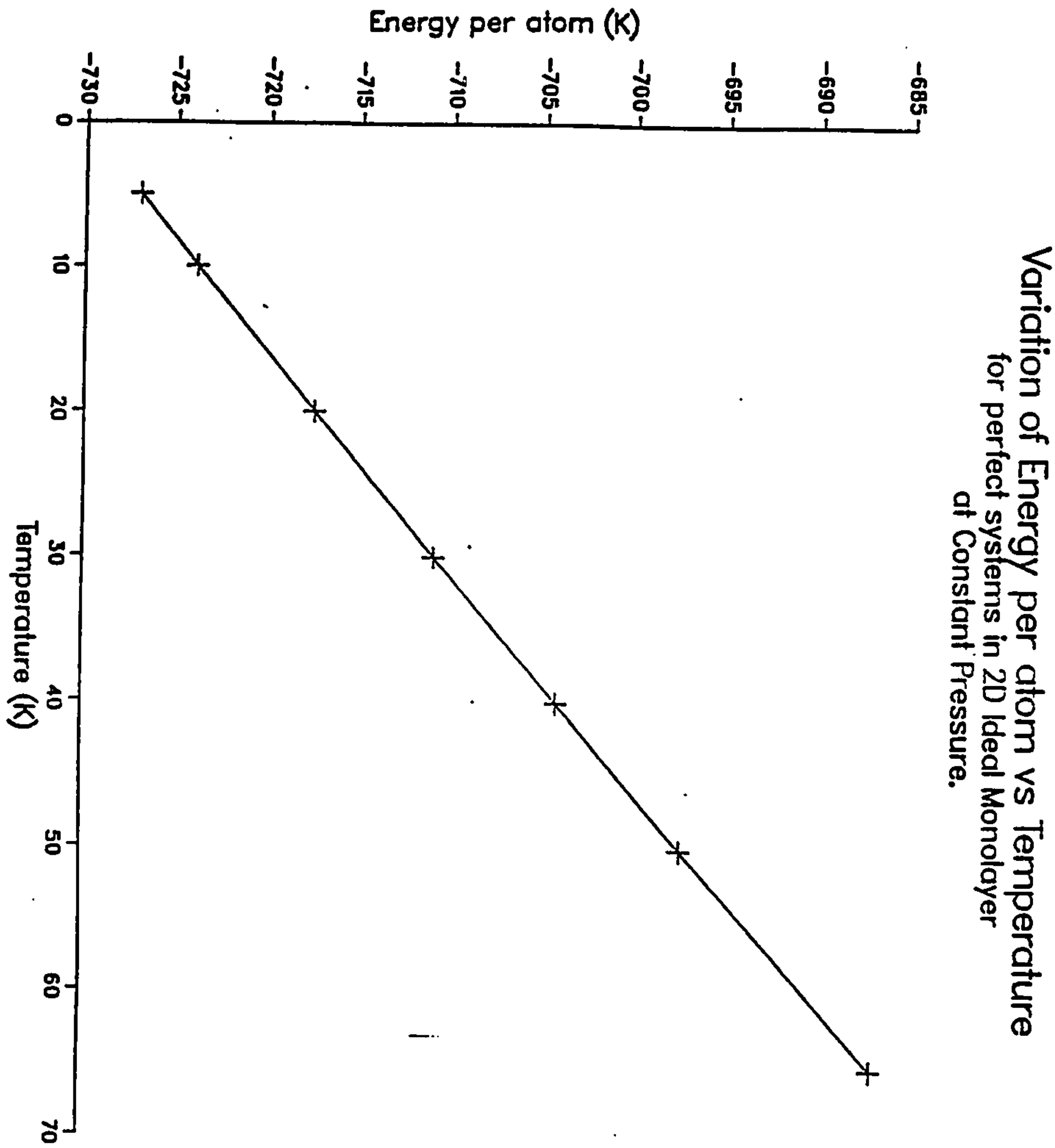

Figure 6.7: Variation of energy by atom with temperature for perfect systems in 2D Xe monolayer, at constant pressure $P = 73.86K/\AA^2$.

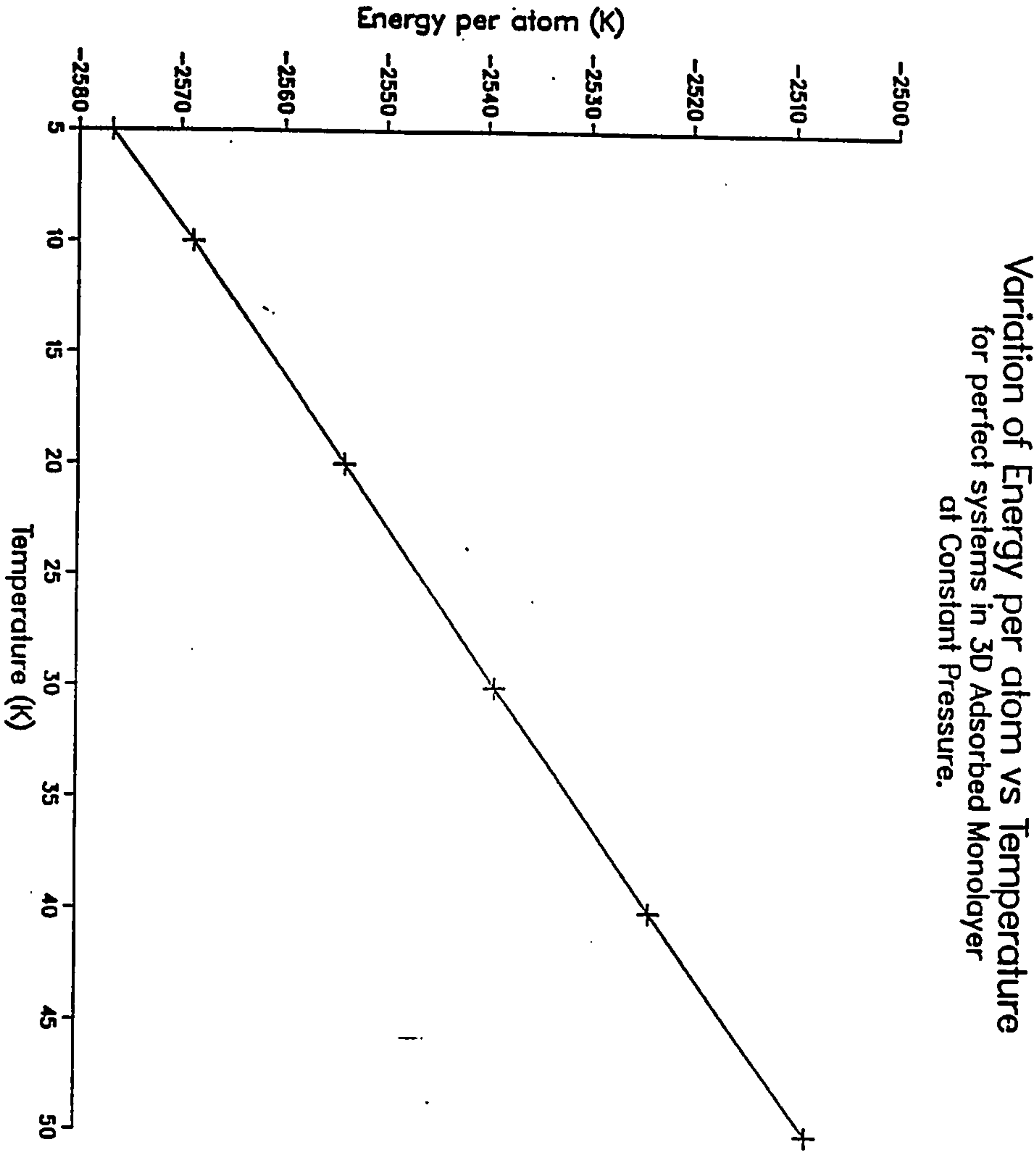

Figure 6.8: Variation of energy by atom with temperature for perfect systems in 3D Xe adsorbed monolayer on Ag(111), at constant pressure $P = 72.34K/\AA^2$.

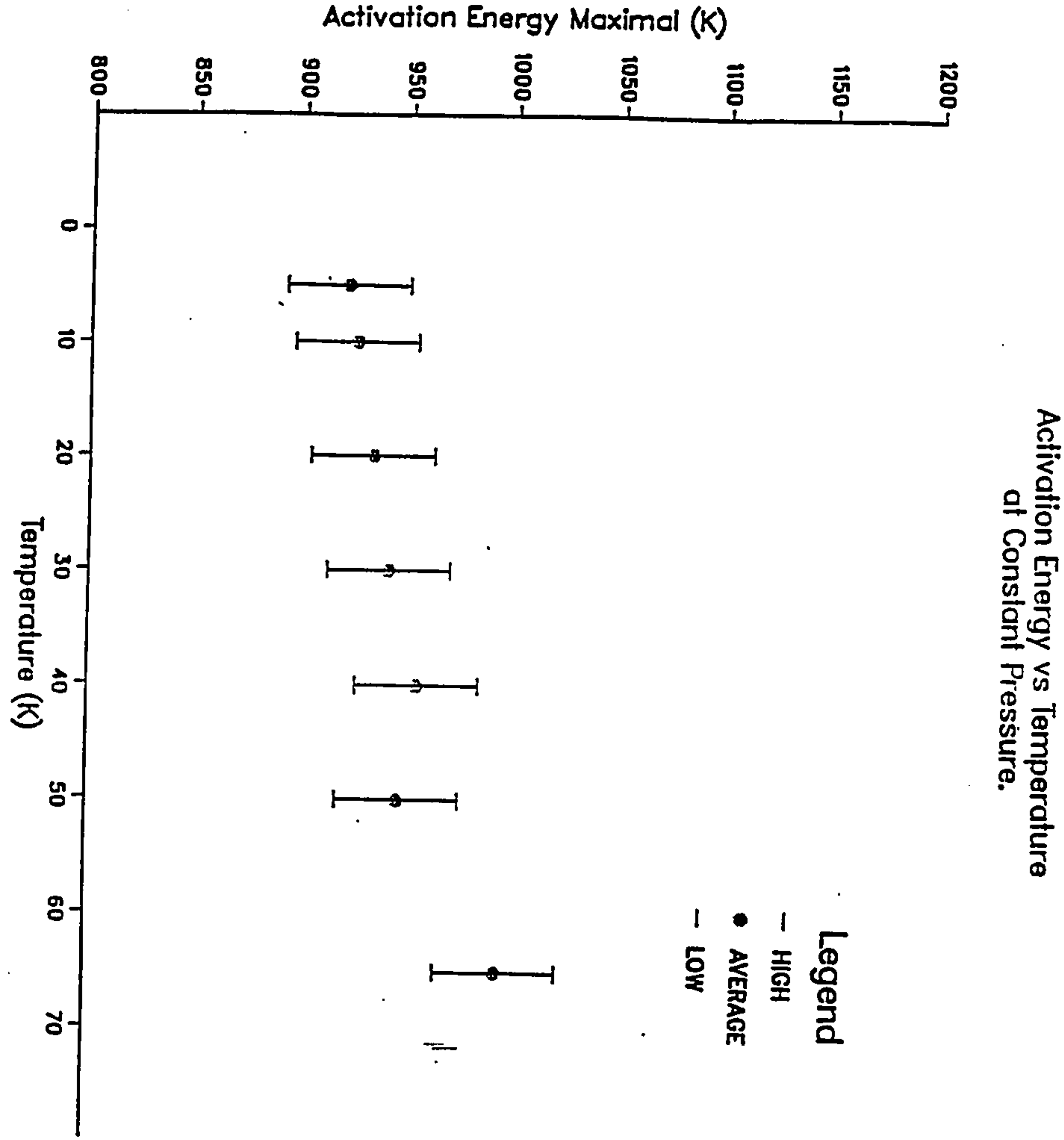

Figure 6.9: Maximum activation energy function of temperature in 2D Xe monolayer, at constant pressure $P = 73.86K/\mathring{A}^2$.

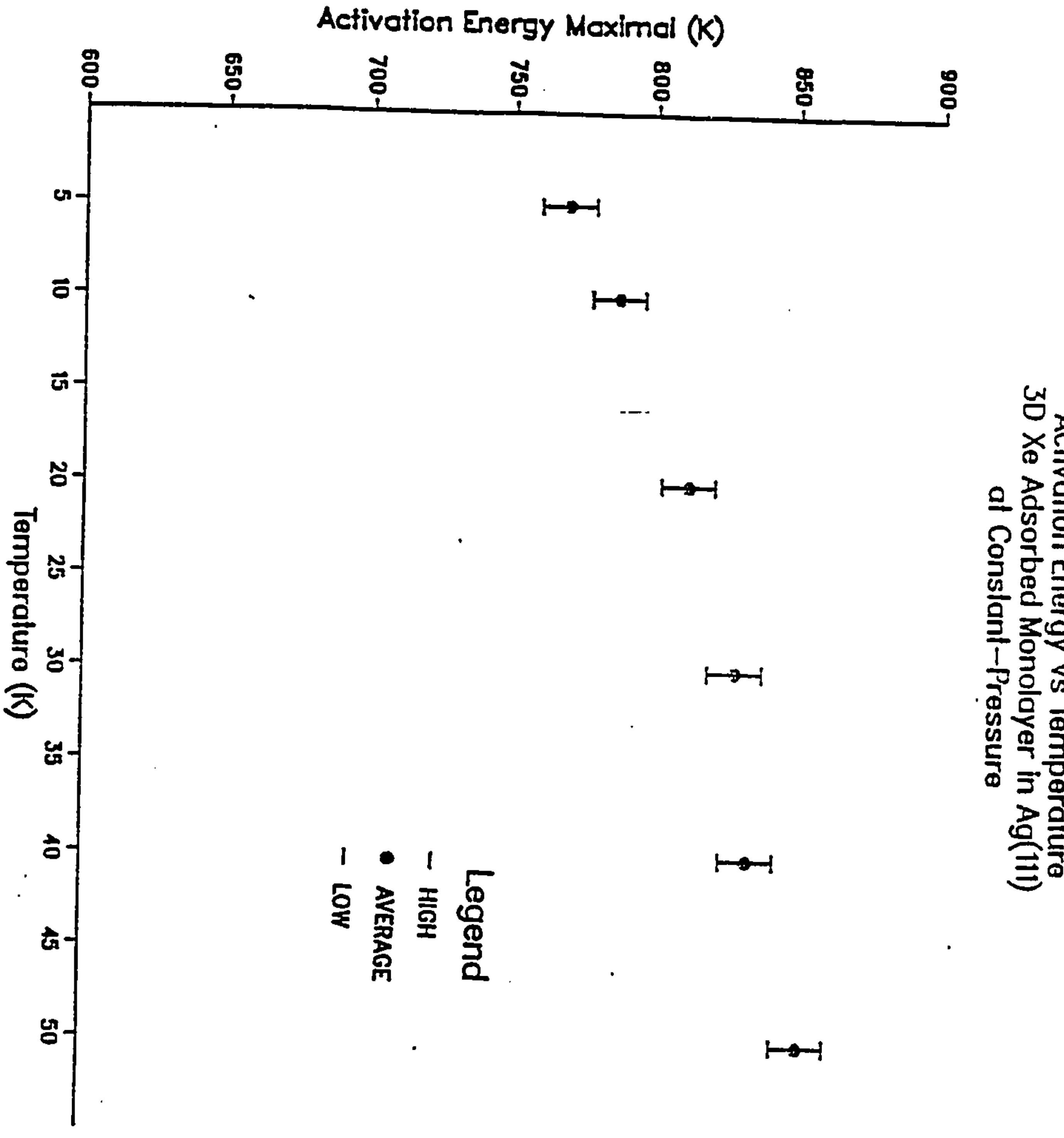

Figure 6.10: Maximum activation energy function of temperature in 3D Xe adsorbed monolayer on Ag(111), at constant pressure $P = 72.34K/\mathring{A}^2$.

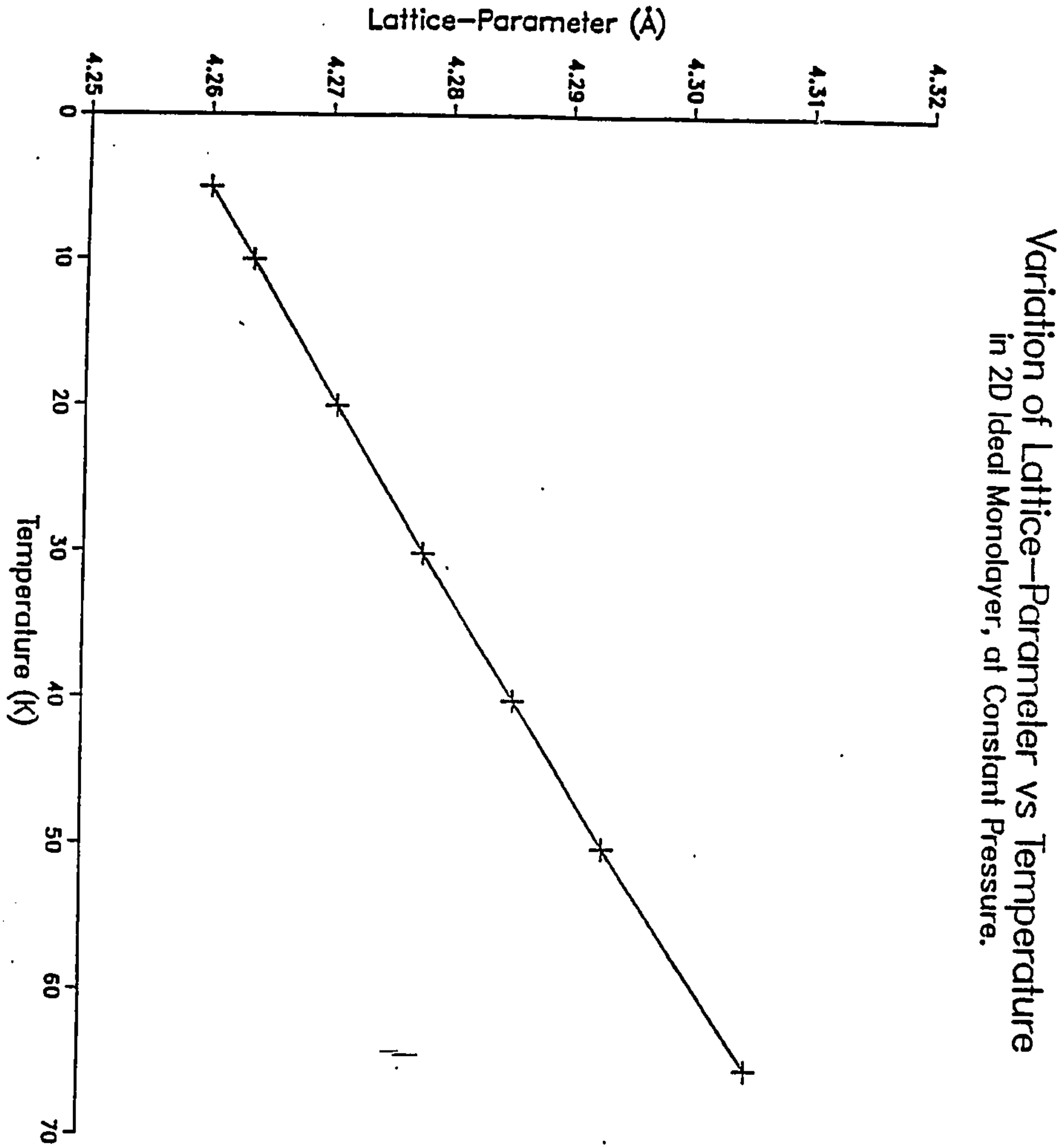

Figure 6.11: Variation of Lattice-parameter with temperature in 2D ideal monolayer, at constant pressure for perfect systems.

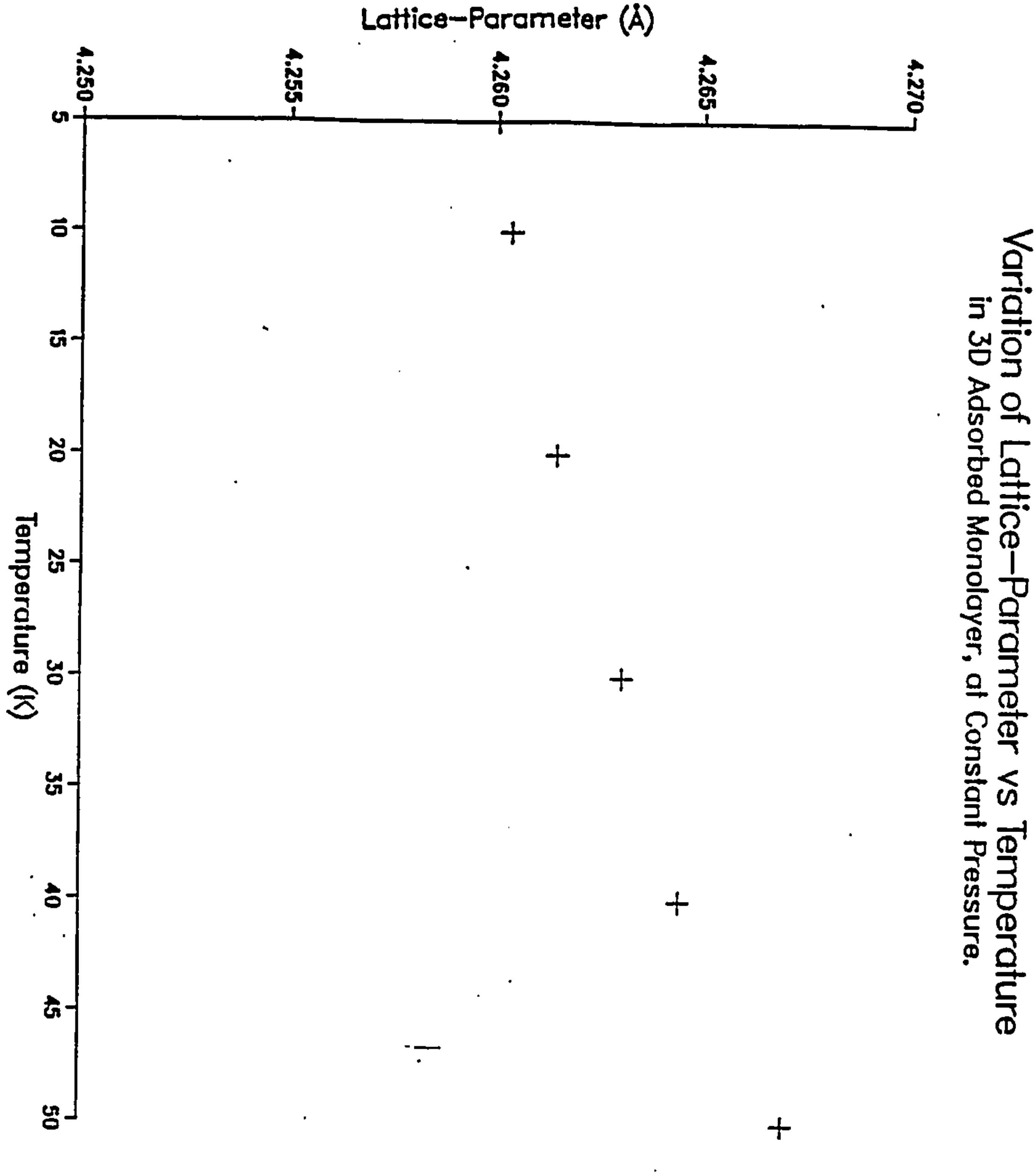

Figure 6.12: Variation of Lattice-parameter with the temperature in 3D Xe adsorbed monolayer on Ag(111), at constant pressure.

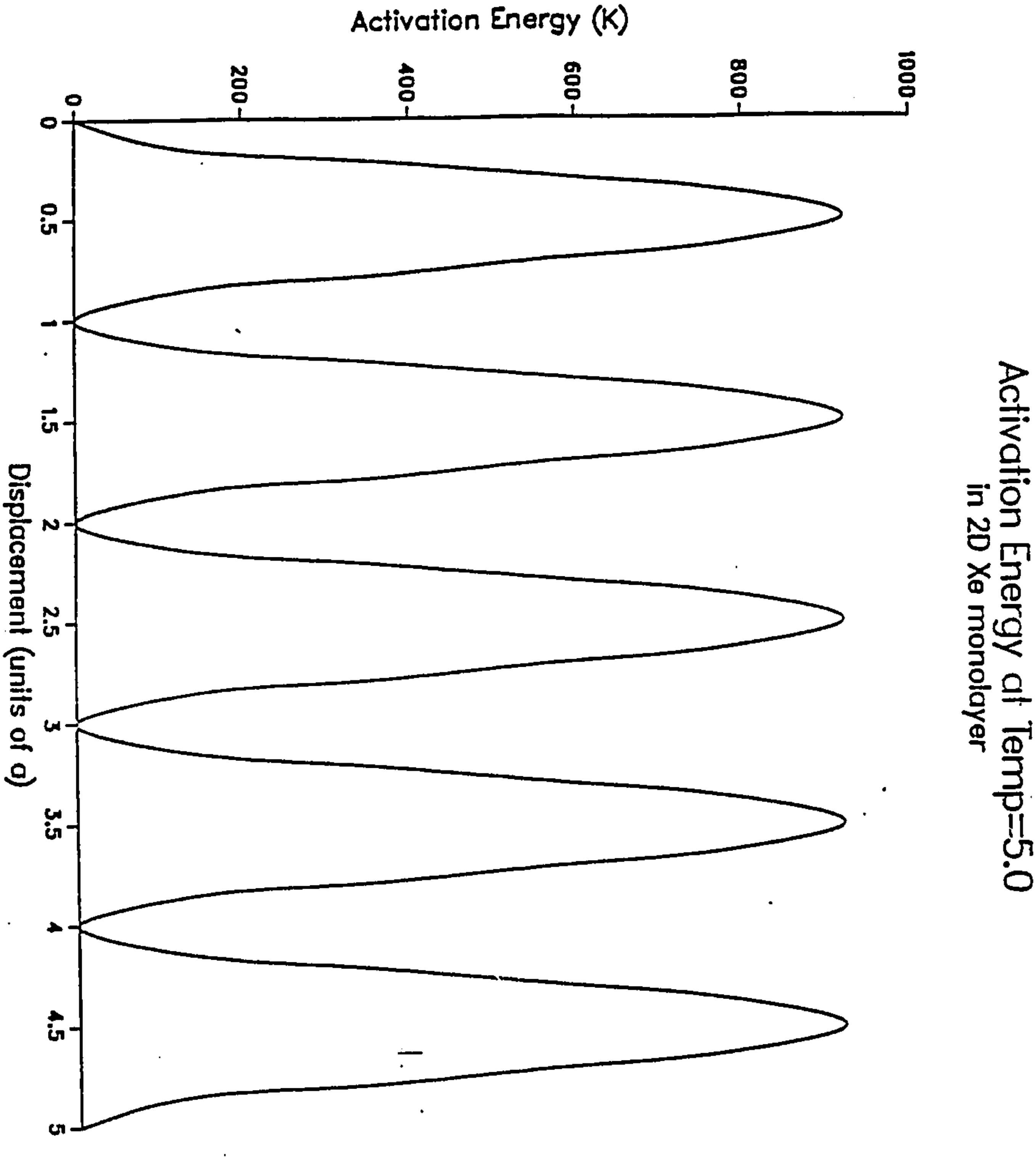

Figure 6.13: Activation Energy transition from site P_2 to site P_1 for five transitions in 2D xenon monolayer.

Chapter 7

INTERACTION BETWEEN TWO VACANCIES IN XENON 2D MONOLAYER

7.1 Introduction:

The interaction between two vacancies in a crystal lattice has been investigated by using the lattice static method [52], considering the effects of the modifications of the forces between a defect and its neighbouring atoms due to the presence of the other defect. This interaction is composed of two parts. The first part is the electronic interaction which occurs if the defects carry an effective charge and is the Coulomb interaction between these effective charges; in metals it is due to the conduction electrons [53]. This part is not always present. The second part is the elastic interaction which takes place through the displacement fields which the defects produce when the lattice is allowed to relax. This second interaction is always present.

The interaction energy between vacancies change as a function of the separation between these two vacancies.

Hardy and Siems [52] [54] show that in an elastically isotropic face-centred cubic lattice, with only nearest neighbour interactions, the two vacancies do have a non-zero interaction energy and this interaction varies as $1/r_s^5$, where r_s is the separation of the vacancies.

Siems and Teodosiu [54] [55] treated other cases of defects with lower symmetry than cubic in an isotropic or anisotropic medium and found that the interaction energy between the two vacancies change as $1/r_s^3$, $1/r_s^4$, corresponding to dipole-multipole interactions.

Eshelby and Tewary [56] [57] treated the case when the elastic constants change around the defects, the interaction energy varies as $1/r_s^6$.

Some of these results were reviewed in 1985 by Vargas [58].

In this work we study the interaction between two vacancies in two dimensional xenon monolayer which can interact directly through a Leonnard Jones potential. The pairwise interaction potentials are assumed to have central symmetry.

7.2 Results and Discussion:

The interaction energy calculations have been done using a NPT molecular dynamics simulation where constant pressure is provided by the algorithm itself, at a finite temperature 5K. The system used is limited to 2601 atoms, the separation betwee two vacancies is expressed in lattice parameter units. The separation is limited to 23a (a: Lattice-parameter), to respect the periodic boundary conditions.

The interaction energy between two vacancies is calculated as

$$E_{Int}(r_S) = E_{2vac}(r_s) - 2\, E_{vac} \tag{7.1}$$

Where $E_{2vac}(r_s)$ is the two vacancies energy formation at different separation of the two vacancies and E_{vac} the vacancy energy formation. $E_{2vac}(r_s)$ and E_{vac} are calculated at constant pressure and constant temperature which is provided by the algorithm itself. For the calculations of $E_{2vac}(r_S)$ and E_{vac} see section 5.2 of chapter 5.

The figure 7.1 shows the variation of the interaction between two vacancies in 2D xenon monolayer. This interaction energy is negative because of the difinition of interaction energy given by equation 7.1, and this energy approaches a minimum value.

The figure 7.2 gives the variation of the logarithmic of the interaction energy and the separation, this shows that there are two different regions of the variation of this interaction energy where the slope is constant. The rate of change of the vacancy interaction energy is equal to -0.56 for small separations and equal to -0.01 for large separations. In view of the size of our system (51x51 atoms) and the expected long range of the vacancy interaction, only the first region is meaningful. The second region reflects multiple interactions between vacancies in the periodic structure.

The very small power observed reflects the long range of the vacancy-vacancy interaction.

The error found for this energy calculations was about ±80 K which is larger then the minimum interaction energy between two vacancies in magnitude, the high values approaches zero.

Our results can not be compared with the results cited in the introduction of this chapter, and this is due to the limitation of the size of our system.

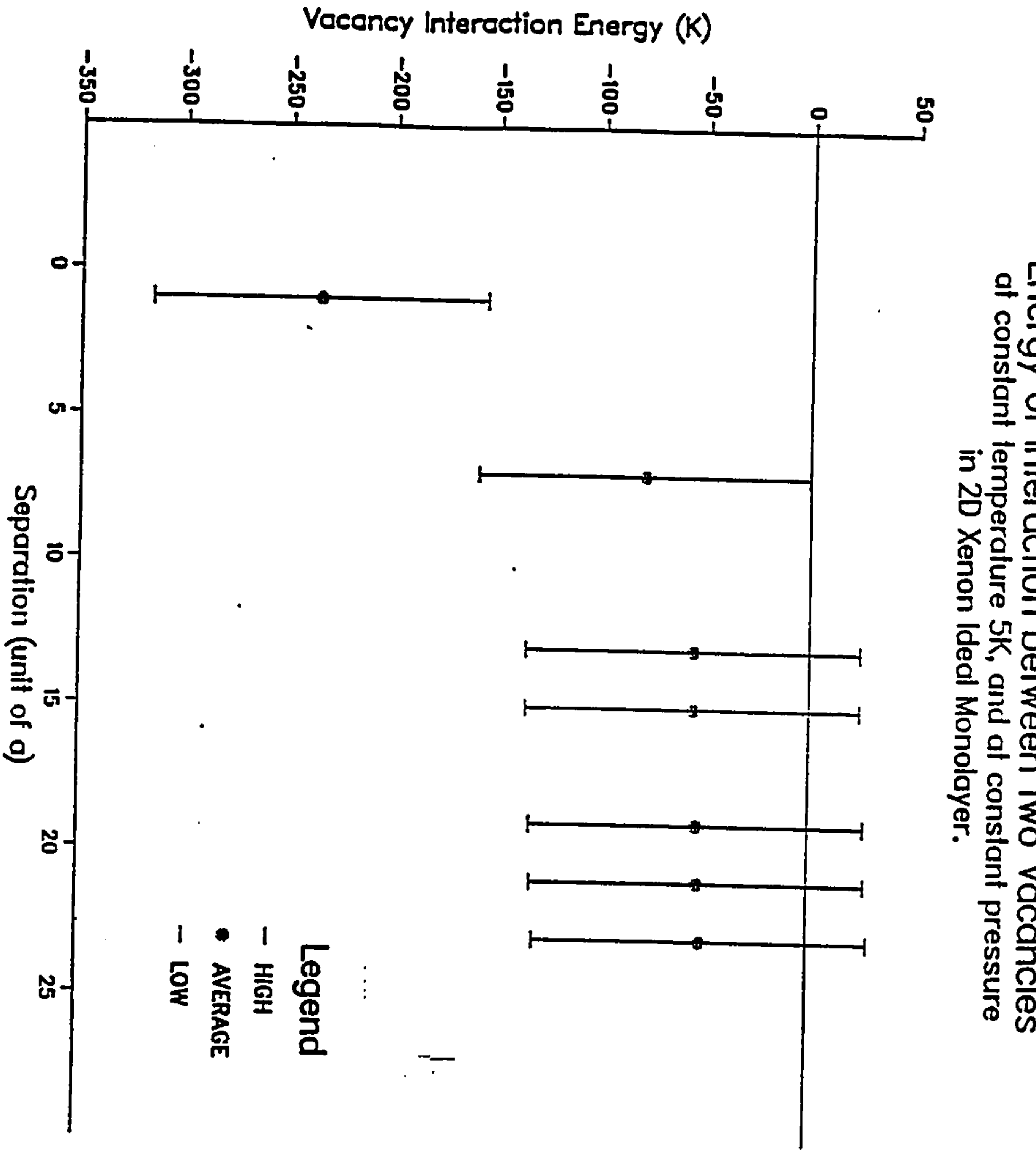

Figure 7.1: Interaction energy between two vacancies in 2D Xe monolayer at temperature of 5K and constant pressure of $74.6K/\mathring{A}^2$ and at lattice parameter of $4.26\mathring{A}$.

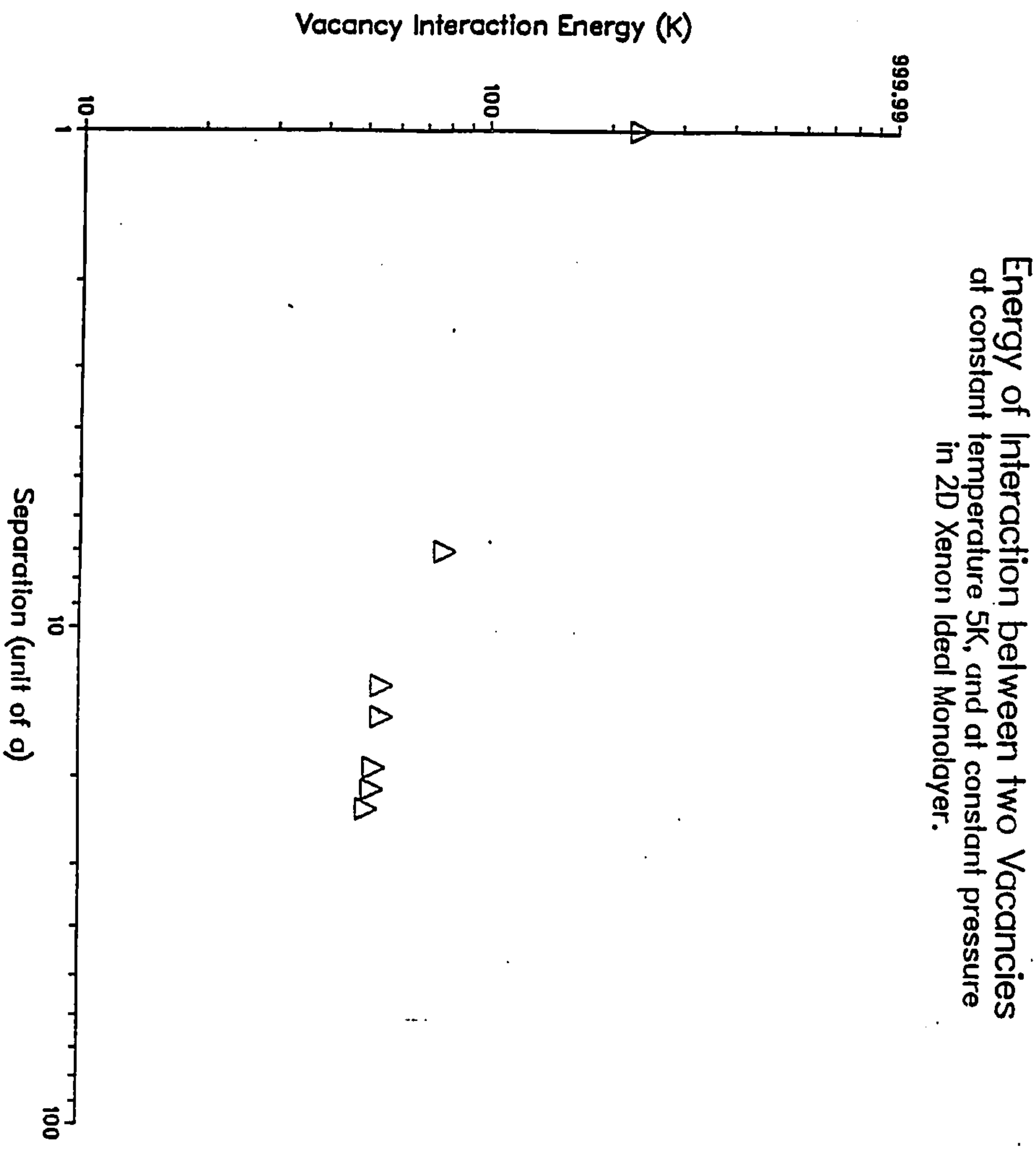

Figure 7.2: Logarithmic variation of interaction energy between 2 vacancies and the separation of these vacancies at temperature of 5K and at constant pressure.

Chapter 8

CONCLUSION:

The principal objective of this work was the calculation of the activation energy of a vacancy with and without the substrate. We find that the presence of the substrate decreases the activation energy.

The activation energy represents 50 percent of the vacancy energy in two dimensions and 41 percent of the vacancy energy in three dimensions at a finite temperature of 5K.

The motion of a vacancy is the result of thermal vibrations in the lattice. In two-dimensions each atom has 2 $k_B\,T$ of thermal energy while in three-dimensions 3 $k_B\,T$. The activation energy goes from $921K$ to $768K$ which is a 20 percent drop, but compared to the thermal energy available at a given temperature this is more like a 45 percent drop in activation energy.
At a temperature of 60 K in two dimensions, the thermal energy of eight atoms is required to make the vacancy move. In three dimensions, it takes only the

thermal energy of four atoms.

In summary it is interesting to note that the activation energy of a vacancy is much smaller than the energy requested to promote the vacancy to the second layer.

The trajectory followed by the activated atom and the rise of the distance of the first nearest-neighbors from the surface of silver(111) in the three dimensional system decreases the activation energy compared to the value found in a two dimensional system.

We observed a small linear increase of the activation energy with the temperature in the range of 5 to 65K. This is an internal energy and not a free energy.
The rate of change of the activation energy with the temperature is greater than that of the energy per atom in two and three dimensions. And smaller than that of the energy of the vacancy in two dimensions.

The last objective was the interaction between two vacancies in a two-dimensional Xe monolayer.
The minimum interaction energy between two vacancies represents about 3 percent of the vacancy energy at a finite temperature of 5K, but is comparable in strength to the interparticle interaction, however much longer ranged.

Vacancies are common in monolayers because thermodynamic equilibrium involves also exchange with the environment which is not the case in bulk solids.

Vacancies are therefore expected to play a significant role in phase changes. The above information can help estimate in particular the importance of the migration of vacancies in a monolayer.

Bibliography

[1] F. Pockels, Nature **43**, 437 (1891); J.W. Strutt, Phil. Mag. **333**, 470 (1892); Lord Rayleigh, Collected Papers, Vol. IV, Cambridge U. P. (1903), p. 425.

[2] R. J. Birgeneau and P.M. Horn, Science **232**, 329 (1986).

[3] R. J. Birgeneau, private communication.

[4] B. Joós and M.S. Duesbery, "self Repair of Monolayers with Vacancy Damage." (preprint) (1992).

[5] R.A. Johnson and E. Brown, Phys. Rev. **127**, 446 (1962) .

[6] K. J. Strandburg, Rev. Mod. Phys. **60**, 161 (1988)

[7] Abraham, F.F., Phys. Rep. Lett. **44**, 463 (1980); Phys. Rep. **80**, 339 (1981a); Phys. Rev. **B23**, 6145 (1981b); Phys. Rev. **B28**, 7338 (1983a); Phys. Rev. Lett. **50**, 978 (1983b); Phys. Rev. **B29**, 2606 (1984).

[8] L. Rayleigh and W. Ramsay, Philos. Trans. R. Soc London Ser. A186, 187 (1895); W.M. Travers, Discovery of the Rare gases, Arnold, London 1928.

[9] M.D. Chinn and S.C. Fain, Jr., Phys. Rev. Lett. **39**, 146 (1977).

[10] J.A. Venables, H.M. Kramer and G.L. Price, Surf. Sci. **55**, 373 (1976).

[11] C.G. Shaw, S.C. Fain, Jr. and M.D. Chinn, Phys. Rev. Lett. **41**, 955 (1978).

[12] G.L Price and J.A. Venables, Surf. Sci. **59**, 509 (1976).

[13] A.D. Novaco and J.P. Mctague, Phys. Rev. Lett. **38**, 1286 (1977).

[14] J. Villarn, Phys. Rev. Lett. **41**, 36 (1978).

[15] M.A. Chesters, M. Hussain and J. Pritchard, Surf. Sci. **35**, 161 (1973).

[16] P.I. Cohen, J. Unguris and M.B. Webb, Surf. Sci. **58** , 429 (1976).

[17] Greiser, N., G.A. Held, R. Frahm, R.L. Greene, P.M. Hornand, R.M. Suter, Phys. Rev. Lett. **59**, 1625 (1987).

[18] Farid F. Abraham, Computational Statistical Mechanics: Methodology, Applications and Supercomputing (1981) p.42.

[19] V. Larher and B. Gilquin, Phys. Rev. **A20**, 1599(1979).

[20] R.A. Aziz, Mol. Phys. **38**, 177 (1979).

[21] B. Joós, B. Bergersen and M.L. Klein, Phys. Rev. **B28**, 7219 (1983).

[22] Gough, D.W., Smith, E.B. and Maitland, G.C., Molec. Phys. **27**, 867 (1974).

[23] Tang, K.T., Norbeck, J.M. and Certain, P.R.,*J*. Chem. Phys., **64**, 3063 (1976).

[24] LeRoy, R.J, *J*. Chem. Phys. **57**, 573 (1972).

[25] L. W. Bruch, Surf. Sci. **125**, 194 (1983).

[26] X-P. Jiang, F. Toigo and M.W. Cole, Surf. Sci.**145**, 281 (1984), **149** (1985).

[27] S. Rauber, J.R. Klein, M.W. Cole and L.W. Bruch, Surf. Sci. **123**, 173 (1982).

[28] N. Esbjerg and J.K. Norskov, Phys. Rev. Lett. **45** ,807 (1980).

[29] B.F. Mason and B.R. Williams, Phys. Rev. Lett. **46** ,1138 (1981).

[30] B.J Alder, T.E. Wainwright, Phys. Rev. **127**, 359 (1962).

[31] Rahman, A., Phys. Rev. **136**, A405 (1964).

[32] Verlet, L., Phys. Rev. **159**, 98 (1967).

[33] Kuchik, J. and B.J. Berne, (1977) in statistical Mecanic B: Time-Dependent Process, edited by B.J. Berne (New York).

[34] Anderson, H.C., *J*. Chem. Phys. **72**, 2384(1980).

[35] Haile, J.M. and Graben, H.W.,*J*. Chem. Phys. **73** 2412 (1980).

[36] B. Joós and M.S. Duesbery, Phys. Rev. **B33**, 8632 (1986).

[37] B. Grossmann, B. Joós and M.S. Duesbery, Phys. Rev. **B39**, 7917 (1989).

[38] D. Brown and J.H.R. Clarke, Mol. Phys. **51**, 1243 (1984).

[39] Hardy, J.R.,.*J* Phys. Chem. Solids, **29**, 2009 (1968).

[40] Beeler, J.R., Jr (1970a) Computer experiments on point defect configurations and energies in Ti-M systems, in: Jaffee, R.J. and N.E. Promisel, The Science Technology and Application of Titanium , Pergamon Press, pp. 383-392.

[41] Beeler, J.R., Jr (1970b) Computer experiments on vacancy migration and ordering kinetics in an AB alloy, in: Kear, B.H., C.T. Sims, N.S. Stoloff and J.H. Westbrook, Ordered Alloys, Clator's Publishing Division, pp. 181-194.

[42] Wilson, W.D. and C.L. Bisson, Rad. Eff. **19**, 53 (1973).

[43] Guinan, M.W., R.N. Stuart and R.J. Borg, Phys. rev. **B15**, 699 (1977).

[44] L.K. Moleko, B. Joós, T.M. Hakim, H.R. Glyde and S.T. Chui, Phys. Rev. **B34**, 2815 (1986).

[45] B. Joós and M.S. Duesbery, Phys. Rev. Lett. **55**, 1997 (1985).

[46] W.G. Hoover, A.J.C. Ladd and N.E. Hoover, in *Interatomic Potentials and Crystalline Defects*, edited by J.K. Lee (Metallurgical Society, Warrendale, Penn., 1981).

[47] K.J. Strandburg, Reviews of Modern Physics, Vol. 60, No. 1 (1988).

[48] M.P. Allen and D.J. Tildesley, Computer Simulation of Liquids (1989).

[49] D.W. Heermann, Computer Simulation Methods (1990)

[50] M. Plischke and B. Bergersen, Equilibrium Statistical Physics (1989)

[51] J.R. Beeler, JR., Radiation Effects Computer Experiments (1983).

[52] Hardy, J.R., Bullough, R.,Phi. Mag. **15**, 237 (1967).

[53] Yamamoto, R.,Phys. Stat. Sol. **74**, 159 (1976).

[54] Siems, R., Phys. Stat. Sol. **30**, 645 (1967).

[55] Teodosiu, C., Elastic Models of Crystal Defects (Berlin: Springer-Verlag).

[56] Eshelby, J.D., Phys. Stat. Sol. **3**, 79 (1956).

[57] Tewary, V.K., Adv. Phys., **23**, 757 (1973).

[58] P. Vargas and H. Kronmüller, Cryst. Latt. Def. and Armorph. Mat., 1985, Vol. 11, pp. 59-72.

Appendix A

VIRIAL EXPRESSION FOR PRESSURE

A quantity of particular interest is the pressure P, which is given by the thermodynamic expression:

$$P = \frac{\partial A}{\partial V} \tag{A.1}$$

Where A is the Helmholtz free energy and is minimized at equilibrium for constant number of atoms N, temperature T and volume V.
The fundamental relationship between statistical mechanics and thermodynamics is given by;

$$A(N,V,T) = -k_B\, T\, \ln Q(N,V,T) \tag{A.2}$$

Where $Q(N,V,T)$ is called **the partition function** defined as:

$$Q(N,V,T) = \frac{1}{N!\, h^{3N}} \int\int d\mathbf{r}\, d\mathbf{p}\, \exp\left[-\frac{1}{k_B\, T}\, \mathcal{H}(\mathbf{r},\mathbf{p})\right] \tag{A.3}$$

Where h is Plancks constant, with generalized coordinates $\mathbf{r} = (\mathbf{r}_1, ..., \mathbf{r}_N)$ in volume element $d\mathbf{r} = d\mathbf{r}_1 ... d\mathbf{r}_N$ and conjugate momenta $\mathbf{p} = (\mathbf{p}_1, ..., \mathbf{p}_N)$ in volume element $d\mathbf{p} = d\mathbf{p}_1 ... d\mathbf{p}_N$. $\mathcal{H}(\mathbf{r}, \mathbf{p})$ is the Hamiltonian.

We can formally integrate out the momentum degrees of freedom, obtaining;

$$Q(N, V, T) = \frac{1}{N!\, h^{3N}} \left[(2\, \pi\, m\, k_B\, T)^{1/2}\right]^{3N}\, Z(N, V, T) \tag{A.4}$$

Where,

$$Z(N, V, T) = \int \exp\left[-\frac{1}{k_B\, T}\, U(\mathbf{r})\right]\, d\mathbf{r} \tag{A.5}$$

$Z(N, V, T)$ is named the configuration integral, $U(\mathbf{r})$ is the potential energy and m is the mass of each atom

From the equations (A.1), (A.2), (A.3), (A.4) and (A.5) we get:

$$P = k_B\, T\, \frac{\partial \ln Q}{\partial V} = k_B\, T\, \frac{\partial \ln Z}{\partial V} \tag{A.6}$$

The virial expression for pressure can be found using the method of Born and Green (1947). For this we have to introduce the following change of variables for the coordinates as:

$$\mathbf{r} = V^{1/3}\, \boldsymbol{\rho} \tag{A.7}$$

Each component of $\boldsymbol{\rho}$ is a dimensionless number between zero and one. This is a very important scaling concept that is exploited in developing constant pressure algorithms for computer simulations.

The equation (A.5) becames;

$$Z(N, V, T) = V^N \int_0^1 d\rho\, \exp\left[-\frac{1}{k_B\, T}\, U(V^{1/3}\, \rho)\right] \tag{A.8}$$

Applying (A.7) to (A.5) and (A.6) yields

$$P\, V = N\, k_B\, T - \frac{1}{3}\, \langle \sum_i \mathbf{r}_i . \nabla_i U(\mathbf{r}_i) \rangle \tag{A.9}$$

Finally;

$$P = \frac{1}{3\,V}\left[3\,N\,k_B\,T - \langle\sum_i \mathbf{r}_i.\nabla_i U(\mathbf{r}_i)\rangle\right] \qquad (A.10)$$

Where the first term $3\,N\,k_B\,T$ represents two times the thermal energy of the system with three degrees of freedom, which is equivalent to the total kinetic energy of our system. The second term is equivalent to the second term of the equation (3.34) of chapter 3.

Appendix B

A PROGRAM OF MOLECULAR DYNAMICS FOR A MONOLAYER OF RARE GAS

PROGRAMME DE DYNAMIQUE MOLÉCULAIRE POUR UNE MONOCOUCHE DE GAS RARE

Le but de ce programme est d'effectuer une simulation obéissant aux règles de la dynamique moléculaire pour une monocouche de gas rare sans et avec substrat.

Le programme contient quelques commentaires d'explications, et il est écrit en FORTRAN .

Une partie de ce programme a été écrite par B. Grossmann .

Liste des sous-programmes:

SETIS : Construction du système et les conditions de Périodicité.
POTEN : Constantes Physiques.
DISVIT : Distribution de vitesse initiale .
SIMTEM : Programme de Simulation Temporelle .
FORCE : Algorithme ' THE SHORT -RANGE FORCE '
FZ : Introduction du substrat 3D .
MOUVE2 : Constante-NVT moléculaire dynamique (Méthode des forces amorties).
MOUVE3 : Constante-NPT moléculaire dynamique (Méthode combinée d'Andersen).
RENEC : Renormalisation de l'énergie .
RECOM : Programme de continuation dans le temps .
STOCKE : Initialisation des atomes de bord et de l'atome activé du centre .
RESET : Sauve-gard des positions des atomes du bord et de l'atome centre .

```
C DICTIONNAIRE DES VARIABLES :
C=============================================================
C
C A1=A2=paramètre de réseau de la cellule dans le système de coordonnées
C      cristallographiques
C X,Y = coordonnées (en Angstroms)
C CX,CY = coordonnées cristallographiques
C VX,VY = vitesses des atomes
C AX,AY = accélération des atomes
C DIS = coordonnées des dislocations
C CDIS = coordonnées cristallographiques des dislocations
C DUX,DUY = déplacements des atomes ,dus aux dislocations, d'après la
C théorie de l'élasticité .
C PMU,PLA,PMU0,PLA0 = coéfficients d'élasticité de Lamé .
C GAMMA = angle de transformation d'un système àl'autre
C NMOLU = nbre de particules dans la céllule unitaire
C LA,LB = nbre de cellules dans une rangée
C NMOL = nbre total d'atomes
C NMOL2 = nbre d'atomes en ne comptant pas les dislocations
C NMOL3 = nbre d'atomes en ne comptant ni les dislocations, ni les
C         atomes situes sur les bords (conditions non-périodiques)
C NDIS = nbre de dislocations
C A1B2,A2B2 = longueur des côtés
C CFDR = facteur de conversion des degrés en radians
C COXY = cos 60
C COYY = sin 60
C OCXY = - cotg 60
C OCYY = - tg 60
C TEMP = température en degré K
C TEMFAC = coefficient servant à maintenir constante la température
C TK = cte de Boltzmann
C DELT = pas temporel utilisé  dans la simulation
C AMASS = masse d'un atome
C VDEL = pas temporel corrigé  en fonction des unités de nos coordonnées
C DELT2 = "                               "
C         et tenant compte de la masse (calcul de ax,ay)
C U,V,W,DU,DV,DW = coéfficients servant à calculer l'énergie et les
C                forces dans notre système  .
C U0,V0,W0,DU0,DV0,DW0 =coéfficients servant à calculer l' énergie et
C                      la force selon la troisième dimension (z)
C DEPLCE= déplacement donné  à l'atome centre .
C-------------------------------------------------------------
```

```
C------------------ PROGRAMME PRINCIPAL ----------------------------------------
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON/DICH/E,E1,E2,SCAL3,SCAL4,NFLAG
      COMMON/DISL/DIS(3,50),CDIS(3,50),NDIS
      COMMON/TEMPER/TEMFAC,TEMP,TK,DELT,AMASS,VDEL,DELT2,DELT3,NTEM,NTEM2,NTEM3
      COMMON/ZUT/A1,SOM,SOU,TSOM,TSOU,GAMMA,LUT,LUT1,LUT2,LA,TEST,TTEST ,LB
      COMMON/PAR/SCAL2,PRES1,PRES2,EK,FM,FM2,PRESS,PRES,ATOT,AREA2, AREA0
     1,AREA,AREA1,AREA3,A,AREAP,AREAS,F,FM3
      COMMON/ITER/NS,NS1,NS2,NPAS,NTEST
      COMMON/RU/IRUN
      COMMON/PR/PRES0,BETA,TMX,TMY,TVX,TVY
C     COMMON/BOUCLE/ZI,ZMOY
      COMMON/VAC/XINT(2601),YINT(2601),CXINT(2601),CYINT(2601),XVAC,
     1YVAC,CXVAC,CYVAC,DEPLCE,MVAC,LVAC
C
      WRITE(7,1)
1     FORMAT(20X,'DEPART FRAIS=0',/,20X,'CONTINUATION=>0')
C
      READ(3,*)NDEP
      READ(3,*) MVAC
      READ(3,*) A1
      READ(3,*) ZI
      READ(3,*) DZ
      READ(3,*) GAMMA
      READ(3,*) LUT
      READ(3,*) LA
      READ(3,*) LB
      READ(3,*) NDIS
      READ(3,*) TEST
      READ(3,*) SOM
      READ(3,*) SOU
      READ(3,*) TTEST
      READ(3,*) TSOM
      READ(3,*) TSOU
      READ(3,*) DELT
      READ(3,*) DELT3
      READ(3,*) NTEM
      READ(3,*) NTEM2
      READ(3,*) NTEM3
      READ(3,*) FM
      READ(3,*) FM2
      READ(3,*) PRES0
      READ(3,*) IRUN
      READ(3,*) TEMP
      READ(3,*) NPAS
C             NFLAG=1
      IF(NDEP.EQ.0) THEN
              CALL SETIS
              ELSE
C             NFLAG=1
```

```
              CALL RECOM
      END IF
      CALL SIMTEM
C       WRITE (3,*) 'principal'
C
      STOP
      END
C=======================================================================
C**********************************************************************
C                                                                     *
C SETIS: Sert  a crée un système périodique .                         *
C Cette subroutine est utilisable pour tout genre de cellule à deux et  trois dimensions .  *
C                                                                     *
C**********************************************************************
C
C================ SETIS   =============================================
C
      SUBROUTINE SETIS
      IMPLICIT REAL*8(A-H,O-Z)
     DIMENSION DMOL(3),R(5000)
     REAL I,K
     COMMON/TEMPER/TEMFAC,TEMP,TK,DELT,AMASS,VDEL,DELT2,DELT3,NTEM,NTEM2,NTEM3
    COMMON/ITER/NS,NS1,NS2,NPAS,NTEST
          COMMON/COOR/X(2601),Y(2601),Z(2601),CX(2601),CY(2601),PPX(2601),PP
     1Y(2601),VX(2601),VY(2601),VZ(2601),AX(2601),AY(2601),AZ(2601),NMOL,NMOL2,NMOL3
    COMMON/ZUT/A1,SOM,SOU,TSOM,TSOU,GAMMA,LUT,LUT1,LUT2,LA,TEST,TTEST ,LB
                                   COMMON/CELL/A1B2,A1B,A2B2,A2B,COXY,COYY,OCXY,OCYY,ALPHA

      COMMON/PAR/SCAL2,PRES1,PRES2,EK,FM,FM2,PRESS,PRES,ATOT,AREA2, AREA0
     1,AREA,AREA1,AREA3,A,AREAP,AREAS,F,FM3
    COMMON/DISL/DIS(3,50),CDIS(3,50),NDIS
    COMMON/DICH/E,E1,E2,SCAL3,SCAL4,NFLAG
     COMMON/PER/PMU,PLA,PMU0,PLA0,IDRA
C    COMMON/BOUCLE/ZI,ZMOY
C
      E1= 1D13
      E2= -1D13
      NTEST=0
      FM3=FM
      BOUCL=0.D0
      A2=A1
      ALPHA=DFLOAT(LA)/DFLOAT(LB)
C      GAMMA=60.D0
C nbre de particules par cellule
      NMOLU=1
C nbre de cellules dans une rangee
      LB=LA
C nbre total de particules
      NMOL=LA*LB*NMOLU
C definition des constantes
      CFDR=DATAN(1.D0)/45.D0
C
      COXY=DCOS(GAMMA*CFDR)
      COYY=DSIN(GAMMA*CFDR)
      OCXY=-COXY/COYY
```

```
      OCYY=1.D0/COYY
C     Ces constantes sont utilisées dans les changements d'un repère à l'autre
C     Construction du systeme
      DSEED=1234567.D0
      NR=NMOL*3
     CALL GGUBS(DSEEED,NR,R)
C
C Indice pour conditions périodiques ou non


     IDRA=1
C mise à zéro des variables de transfert
      L=0
      M=0
      N=0
      K=1
C      WRITE(3,*) ZI,ZMOY,NMOL3,NS,NFLAG
        NVAC1=NMOL/2-(NVAC-1)*LA+1
       NVAC2=NMOL/2+(NVAC-1)*LA+1
      DO 16 KA=0,LA-1
          DMOL(1)=A1*DFLOAT(KA)
          DO 17 KB=0,LB-1
              DMOL(2)=A2*DFLOAT(KB)
              IF (NDIS.EQ.0) GOTO 1
             IF (NDIS.EQ.1) GOTO 5
              IF (NDIS.EQ.2) GOTO 9
C Introduction des dislocations
                DO 18 O=NMOL/2-NDIS/2+1,NMOL/2+NDIS/2+1
                  IF(O.EQ.M+1) THEN
                          WRITE(7,*) 'o=m'
                          CDIS(1,L+1)=DMOL(1)
                          CDIS(2,L+1)=DMOL(2)
                          CDIS(3,L+1)=DFLOAT(M+1)
                          CX(M+1)=0.D0
                          CY(M+1)=0.D0
C                          Z(M+1)=0.D0
C
                          L=L+1
                          M=M+1
C                           WRITE(3,*) M,CX(M),CY(M)
                          GOTO 17
                  ENDIF
 18             CONTINUE
 5          IF(NDIS.EQ.1.AND.M+1.EQ.NMOL/2+1) THEN
                          CDIS(1,L+1)=DMOL(1)
                          CDIS(2,L+1)=DMOL(2)
                          CDIS(3,L+1)=DFLOAT(M+1)
                          CX(M+1)=0.D0
                          CY(M+1)=0.D0
C                          Z(M+1)=0.D0
                          M=M+1
                          L=L+1
                          GOTO 17
                 ENDIF
            IF(NDIS.EQ.2.AND.M+1.EQ.NVAC1)  GOTO 6
            IF(NDIS.EQ.2.AND.M+1.EQ.NVAC2)  GOTO 6
```

```
           GOTO 1
                        CDIS(1,L+1)=DMOL(1)
                        CDIS(2,L+1)=DMOL(2)
                        CDIS(3,L+1)=DFLOAT(M+1)
                        CX(M+1)=0.D0
                        CY(M+1)=0.D0
C                        Z(M+1)=0.D0
                        M=M+1
                        L=L+1


                        GOTO 17
 1             CX(M+1)=DMOL(1)
               CY(M+1)=DMOL(2)
               IF(NFLAG.EQ.0) THEN
C                         Z(M+1)=ZI+DZ*(2*R(K)-1)
                        K=K+1
                       ELSE
C                        Z(M+1)=ZMOY/DFLOAT(NMOL3)/DFLOAT(NS)
               ENDIF
               M=M+1
               N=N+1
17          CONTINUE
16       CONTINUE
      NMOL2=N
      NDIS=L
      A1B2=DFLOAT(LA)*A1
      A1B=A1B2/2.D0
      A2B2=DFLOAT(LB)*A2
      A2B=A2B2/2.D0
C
C calcul des centres des dislocations
      IF(NDIS.EQ.0) GOTO 2
       CDIS(1,1)=(CDIS(1,1)+CX(DINT(CDIS(3,1))-1))/2.D0
       CDIS(2,1)=(CDIS(2,1)+CY(DINT(CDIS(3,1))-1))/2.D0
       CDIS(1,NDIS)=(CDIS(1,NDIS)+CX(DINT(CDIS(3,NDIS))+1))/2.D0
       CDIS(2,NDIS)=(CDIS(2,NDIS)+CY(DINT(CDIS(3,NDIS))+1))/2.D0
C
C on calcule le centre géometrique de la boite
C
 2     DO 181 J=1,3
          DMOL(J)=0.D0
 181   CONTINUE
      DO 182 M=1,NMOL
          DMOL(1)=DMOL(1)+CX(M)
          DMOL(2)=DMOL(2)+CY(M)
 182   CONTINUE
CC       WRITE(9,*)'NMOL2=',NMOL2
      DMOL(1)=DMOL(1)/DFLOAT(NMOL2)
      DMOL(2)=DMOL(2)/DFLOAT(NMOL2)
C On déplace l'origine au centre de la boite
      DO 20 M=1,NMOL
         IF(CX(M).EQ.0.D0.AND.CY(M).EQ.0.D0.AND.M.NE.1) GOTO 20
         CX(M)=CX(M)-DMOL(1)
         CY(M)=CY(M)-DMOL(2)
C transformation du repère cristallographique au repère orthogonal
```

```
          X(M)=CX(M)+COXY*CY(M)
          Y(M)=     COYY*CY(M)
 20   CONTINUE
        DO 31 I=1,NMOL
           IF(NDIS.NE.O.AND.I.EQ.NMOL/2+1) GOTO 31
          WRITE(17,*) X(I),Y(I)
 31   CONTINUE
C
C
     IF(NDIS.EQ.0) GOTO 3
     DO 21 M=1,NDIS


       CDIS(1,M)=CDIS(1,M)-DMOL(1)
          CDIS(2,M)=CDIS(2,M)-DMOL(2)
C meme chose pour les dislocations
          DIS(1,M)=CDIS(1,M)+COXY*CDIS(2,M)
          DIS(2,M)=         COYY*CDIS(2,M)
          DIS(3,M)=CDIS(3,M)
 21   CONTINUE
C
 3     N2=NMOL-NMOL2
C
C
C     WRITE (3,*) 'setis'
C
    CALL POTEN
 C
    CALL DISVIT
 C
 C
 C
    RETURN
     END
C===========================================================================
C
C
C*******************************************************************
C POTEN :                                                           *
C Introdruire les Constantes Physiques .                            *
C*******************************************************************
C
C==========================POTEN ==========================================
C
C
    SUBROUTINE POTEN
     IMPLICIT REAL*8(A-H,O-Z)
C
COMMON/TEMPER/TEMFAC,TEMP,TK,DELT,AMASS,VDEL,DELT2,DELT3,NTEM,NTEM2,NTEM3
COMMON/COOR/X(2601),Y(2601),Z(2601),CX(2601),CY(2601),PPX(2601),PP
1Y(2601),VX(2601),VY(2601),VZ(2601),AX(2601),AY(2601),AZ(2601),NMOL,NMOL2,NMOL3
COMMON/ZUT/A1,SOM,SOU,TSOM,TSOU,GAMMA,LUT,LUT1,LUT2,LA,TEST,TTEST,LB

C
C constantes physiques
```

```
C
      AVOGN=6.022045D3
C  AVOGN EST MUTIPLIE PAR 1.D20!
      TK=1.38066D-16
C TK EN (ERG/KELVIN)!
      AMASS=131.3D0
C
      AMASS=AMASS/AVOGN
C Amass doit etre multiplie par 1.D-20!
C 2 dimensions=1.d0       3 dimensions=3.d0/2.d0
      TEMFAC=1.D0*TEMP*(TK*1.D20)
C          WRITE(3,*) TEMFAC
C TEMFAC en erg E+20!
C PAS TEMPOREL


    VDEL=1.D8*DELT
      DELT2=(DELT*1.D28)/AMASS
C      WRITE (3,*) 'poten'
C
      RETURN
      END
C=====================================================================================
C
C
C.....................................................................................
C DISTRIBUTIONS DES VITESSES :
C  Construction du système  (CM/S)
C.....................................................................................
C
C================== DISTRIBUTION DES VITESSES ====================================
C
C
      SUBROUTINE DISVIT
      IMPLICIT REAL*8(A-H,O-Z)
C
     DIMENSION VB(3),R(5202)
     COMMON/ITER/NS,NS1,NS2,NPAS,NTEST
        COMMON/UL/VPX(2601),VPY(2601),VPZ(2601),P1X(2601),P1Y(2601),P2X(26
     101),P2Y(2601),PX(2601),PY(2601),TRX(2601),TRY(2601),XP(2601), YP(2601),P(100)
        COMMON/COOR/X(2601),Y(2601),Z(2601),CX(2601),CY(2601),PPX(2601),PP
     1Y(2601),VX(2601),VY(2601),VZ(2601),AX(2601),AY(2601),AZ(2601),NMOL
     1,NMOL2,NMOL3
     COMMON/TEMPER/TEMFAC,TEMP,TK,DELT,AMASS,VDEL,DELT2,DELT3,NTEM,NTEM
     12,NTEM3
     COMMON/DISL/DIS(3,50),CDIS(3,50),NDIS
     COMMON/PER/PMU,PLA,PMU0,PLA0,IDRA
      COMMON/ZUT/A1,SOM,SOU,TSOM,TSOU,GAMMA,LUT,LUT1,LUT2,LA,TEST,TTEST
     1,LB
      COMMON/PAR/SCAL2,PRES1,PRES2,EK,FM,FM2,PRESS,PRES,ATOT,AREA2, AREA0
     1,AREA,AREA1,AREA3,A,AREAP,AREAS,F,FM3
C
      EK=0.D0
      DO 3 J=1,2
         VB(J)=0.D0
```

```
3      CONTINUE
C Vecteurs vitesse en CM/S!
      DSEED=1234576.D0
      NR=NMOL*3
      CALL GGUBS(DSEED,NR,R)
      K=1
      DO 1 M=1,NMOL
         IF(NDIS.NE.0) THEN
                    DO 2 N=1,NDIS
                       IF(DFLOAT(M).EQ.DIS(3,N)) GOTO 1
2                   CONTINUE
                    IF(IDRA.NE.0) GOTO 7
                    IF(M.LE.LA) GOTO 1
                    IF(M.GT.(NMOL-LA)) GOTO 1
                    IF(MOD(M,LA).EQ.0) GOTO 1



                    IF(MOD(M,LA).EQ.1) GOTO 1
 7         ENDIF
           VB(1)=VB(1)+R(K)*2.D0-1.D0
           VB(2)=VB(2)+R(K+1)*2.D0-1.D0
C           VB(3)=VB(3)+R(K+2)*2.D0-1.D0
           TRX(M)=R(K)*2.D0-1.D0
           TRY(M)=R(K+1)*2.D0-1.D0
C           VZ(M)=R(K+2)*2.D0-1.D0
          K=K+1
 1    CONTINUE
      NMOL3=K-1
C
COn  soustrait la vitesse de la boite  .
C
C Ne pas oublier vz
      DO 4 J=1,2
           VB(J)=VB(J)/DFLOAT(NMOL3)
 4    CONTINUE
C
      DO 5 M=1,NMOL
         IF(NDIS.NE.0) THEN
                    DO 11 N=1,NDIS
                       IF(DFLOAT(M).EQ.DIS(3,N)) GOTO 5
 11                 CONTINUE
                    IF(IDRA.NE.0) GOTO 8
                    IF(M.LE.LA) GOTO 5
                    IF(M.GT.(NMOL-LA)) GOTO 5
                    IF(MOD(M,LA).EQ.0) GOTO 5
                    IF(MOD(M,LA).EQ.1) GOTO 5
 8         ENDIF
           TRX(M)=TRX(M)-VB(1)
           TRY(M)=TRY(M)-VB(2)
C           VZ(M)=VZ(M)-VB(3)
C Doit-on tenir compte de vz dans le calcul du facteur de pondération de l' énergie ?
            EK=EK+TRX(M)**2+TRY(M)**2
5     CONTINUE
      EK=EK*AMASS/DFLOAT(NMOL3)*.5D0
```

```
C       WRITE(9,111) EK
 111      FORMAT(1X,'EK PAR ATOME=',1PE15.8)
      IF(NS.EQ.1) THEN
              WRITE(7,*) EK
              ENDIF
C EK EN (CM/S)**2*GRAMS E+20=ERGS E+20!
      SCALE=DSQRT(TEMFAC/EK)
C           WRITE(3,*) SCALE
C SCALE est un nombre pur!
      DO 10 M=1,NMOL
         IF(NDIS.NE.0) THEN
                    DO 6 N=1,NDIS
                       IF(DFLOAT(M).EQ.DIS(3,N)) GOTO 10
 6                  CONTINUE
                    IF(IDRA.NE.0) GOTO 9
                    IF(M.LE.LA) GOTO 10
                    IF(M.GT.(NMOL-LA)) GOTO 10
                    IF(MOD(M,LA).EQ.0) GOTO 10


                 IF(MOD(M,LA).EQ.1) GOTO 10
 9         ENDIF
           TRX(M)=TRX(M)*SCALE
           TRY(M)=TRY(M)*SCALE
C           VZ(M)=VZ(M)*SCALE
10     CONTINUE
C       WRITE (3,*) 'disvit'
      RETURN
      END
C======================================================================
C
C
C*********************************************************************
C                                                                    *
C SIMULATION TEMPORELLE: Ce sous programme effectue la boucle sur    *
C le temps.                                                          *
C                                                                    *
C*********************************************************************
C
C====================   SIMULATION TEMPORELLE  ========================
C
C
      SUBROUTINE SIMTEM
       IMPLICIT REAL*8(A-H,O-Z)
        REAL*4 TE0,TE1,TE2,TE3,TE4,TE5,TE6,TE7,TE8,TE9,T1,T2,T3,T4
      INTEGER MO
      LOGICAL*1 DAT(23)
C
        COMMON/UL/VPX(2601),VPY(2601),VPZ(2601),P1X(2601),P1Y(2601),P2X(2
             1601),P2Y(2601),PX(2601),PY(2601),TRX(2601),TRY(2601),XP(2601),    YP(2601),P(100)

     COMMON/ZUT/A1,SOM,SOU,TSOM,TSOU,GGAMMA,LUT,LUT1,LUT2,LA,TEST,TTEST,LB
    COMMON/ITR/NS,NS1,NS2,NPAS,NTEST
        COMMON/COOR/X(2601),Y(2601),Z(2601),CX(2601),CY(2601),PPX(2601),PP
     1Y(2601),VX(2601),VY(2601),VZ(2601),AX(2601),AY(2601),AZ(2601),NMOL,NMOL2,NMOL3
       COMMON/APPROX/U(132),V(132),W(132),DU(132),DV(132),DW(132),U0(1
```

```
     128),V0(128),W0(128),DU0(128),DV0(128),DW0(128)
     COMMON/TEMPER/TEMFAC,TEMP,TK,DELT,AMASS,VDEL,DELT2,DELT3,NTEM,NTEM2,NTEM3
     COMMON/DISL/DIS(3,50),CDIS(3,50),NDIS
     COMMON/DICH/E,E1,E2,SCAL3,SCAL4,NFLAG
     COMMON/PAR/SCAL2,PRES1,PRES2,EK,FM,FM2,PRESS,PRES,ATOT,AREA2, AREA0
    1,ARE/,AREA1,AREA3,A,AREAP,AREAS,F,FM3
     COMMON/RU/IRUN
     COMMON/DRA/L
     COMMON/PR/PRES0,BETA,TMX,TMY,TVX,TVY
C    COMMON/BOUCLE/ZI,ZMOY
      COMMON/VAC/XINT(2601),YINT(2601),CXINT(2601),CYINT(2601),XVAC,
    1YVAC,CXVAC,CYVAC,DEPLCE,MVAC,LVAC
C
      MO=DINT(1.D-17/DELT/(DSQRT(3*TK*TEMP/AMASS)))
      DELT=(1.D-20/(DSQRT(3*TK*TEMP/AMASS)))
     WRITE(12,*) 'DELT=',DELT
     WRITE(12,*) 'MO=',MO
     DELT3=DELT
     VDEL=1.D8*DELT
     DELT2=(DELT*1.D28)/AMASS


     WRITE (3,*) D
     JIDRA=0
     L=0
C     IF(NFLAG.EQ.1.AND.IRUN.EQ.0) GOTO 999
C
C
C Ce sous programme effectue la boucle sur le temps.
C
     WRITE(7,888) NPAS
888   FORMAT(10X,'NOMBRE DE PAS=',I4)
C     READ(3,*)NPAS
C
     WRITE(7,777)
777   FORMAT(1X,'NSAVEL=0 "PAS DE MISE EN MEMOIRE"; =>1"MISE EN MEMOIR
    1E DES POSITIONS FINALES."')
     OPEN (UNIT=4,FILE='XENAN')
     READ(4,*) U,V,W,DU,DV,DW
     CLOSE (UNIT=4)
       WRITE(11,*) 'FM3=',FM3
       WRITE(11,*) 'MVAC=',MVAC
        IF(MVAC.NE.0) THEN
          CALL STOCKE
        ENDIF
C
999   IDIM=23
     T1=0.D0
     T2=0.D0
     T3=0.D0
     TE0=0.D0
     TE1=0.D0
     TE2=0.D0
     TE3=0.D0
     TE4=0.D0
     TE5=0.D0
```

```
      TE7=0.D0
      TE8=0.D0
      TE9=0.D0
      CALL DATETM(DAT,IDIM,TE1,TE2,TE0)
      DO 1000 NS=1+IRUN,NPAS+IRUN
        WRITE(3,*) 'NS=',NS ,'FM3=',FM3
      T1=0.D0
      T2=0.D0
      T3=0.D0
      TE0=0.D0
      TE1=0.D0
      TE2=0.D0
      TE3=0.D0
      TE4=0.D0
      TE5=0.D0
      TE7=0.D0
          TE8=0.D0
          TE9=0.D0
C
          BETA=1.D0
          CALL DATETM(DAT,IDIM,TE1,TE2,TE3)
          IF(MOD(NS,LUT).EQ.0) CALL RENEC


        CALL DATETM(DAT,IDIM,TE1,TE2,TE4)
          T1=T1+TE4-TE3
C
          IF(TEST.EQ.1.D0) THEN
                 IF(NS.GT.IRUN.AND.NS.LE.500+IRUN) THEN
                            PRES0=PRES0-SOU
                            ATOT=0.D0
                            PRESS=0.D0
                            A=0.D0
                            NTEST=0
                            NTEM3=0
                            E1= 1D13
                            E2= -1D13
                            FM3=FM
                            F=0.D0
                 ENDIF
          ENDIF
          IF(TEST.EQ.0.D0) THEN
                 IF(NS.GT.IRUN.AND.NS.LE.500+IRUN) THEN
                            PRES0=PRES0+SOM
                            ATOT=0.D0
                            PRESS=0.D0
                            A=0.D0
                            NTEST=0
                            NTEM3=0
                            E1= 1D13
                            E2= -1D13
                            FM3=FM
                            F=0.D0
                 ENDIF
          ENDIF
C
```

```
        IF(TTEST.EQ.1.D0) THEN
                 IF(NS.GT.IRUN.AND.NS.LE.200+IRUN) THEN
                     DELT=(1.D-20/(DSQRT(3*TK*TEMP/AMASS)))
                  DELT3=DELT
                   VDEL=1.D8*DELT
                    DELT2=(DELT*1.D28)/AMASS
                  TEMP=TEMP-TSOU
                  ENDIF
        ENDIF
        IF(TTEST.EQ.0.D0) THEN
                 IF(NS.GT.IRUN.AND.NS.LE.200+IRUN) THEN
                     DELT=(1.D-20/(DSQRT(3*TK*TEMP/AMASS)))
                  DELT3=DELT
                   VDEL1=1.D8*DELT
                    DELT2=(DELT*1.D28)/AMASS
                  TEMP=TEMP+TSOM
                  ENDIF
        ENDIF
C
C  Boucle Temporelle .
         IF(MVAC.NE.0) CALL RESET
         CALL DATETM(DAT,IDIM,TE1,TE2,TE5)
        CALL FORCE
C        CALL FZ


        CALL DATETM(DAT,IDIM,TE1,TE2,TE6)
         T2=T2+TE6-TE5
C
C
         CALL DATETM(DAT,IDIM,TE1,TE2,TE7)
        CALL MOUVE2
         CALL DATETM(DAT,IDIM,TE1,TE2,TE9)
         T3=T3+TE9-TE7
C       WRITE(3,*) 'MVAC=',MVAC
C
 1000 CONTINUE
        WRITE(11,*) 'FM3=',FM3
        WRITE(11,*) 'MVAC=',MVAC
C     T5=TE11-TE0
      T4=TE9-TE0
      WRITE(7,*) ' temps renec =',T1
      WRITE(7,*) ' temps force =',T2
      WRITE(7,*) 'temps mouve2 =',T3
C      WRITE(7,*) 'temps reset  =',T4
      WRITE(7,*) 'temps  total =',T4
              OPEN (UNIT=10)
      IF(MVAC.NE.0) CALL RESET
C Sauvegarder les coordonnés des bords et de l' atome vacante.
              DO 111 M=1,NMOL
              IF(NDIS.NE.0.AND.M.EQ.NMOL/2+1) GOTO 111
                   WRITE(10,7880) X(M),Y(M)
 7880                FORMAT(1X,D15.8,2X,D15.8)
 111            CONTINUE
               CLOSE (UNIT=10)
              NMOL2 =NMOL-NDIS
```

```
                  OPEN(UNIT=30)
                  WRITE(30,*) MVAC
                  WRITE(30,*) LA
                  WRITE(30,*) LB
                  WRITE(30,*) NMOL3
                  WRITE(30,*) NMOL2
                  WRITE(30,*) NDIS
                  WRITE(30,*) A1
                  WRITE(30,*) AREA
C                 WRITE(30,*) AREA1
C                 WRITE(30,*) AREA2
                  WRITE(30,*) NTEM3
                  WRITE(30,*) NTEST
                  WRITE(30,*) F
                  WRITE(30,*) ATOT
                  WRITE(30,*) PRESS
                  WRITE(30,*) A
                  WRITE(30,*) E1
                  WRITE(30,*) E2
                  WRITE(30,*) TMX
                  WRITE(30,*) TMY
                  WRITE(30,*) TVX
                  WRITE(30,*) TVY
                  DO 119 M=1,NMOL
                     WRITE(30,7882)P1X(M),P1Y(M),P2X(M),P2Y(M)
 7882                 FORMAT(1X,D15.8,1X,D15.8,1X,D15.8,1X,D15.8)


 119                CONTINUE
C                 DO 120 M=1,NMOL
C                        WRITE(30,7882) P2X(M),P2Y(M)
C120                CONTINUE
                  DO 121 M=1,NMOL
                     WRITE(30,7883) TRX(M),TRY(M),VX(M),VY(M)
C   1                                          ,VZ(M)
 7883                  FORMAT(1X,D15.8,1X,D15.8,1X,D15.8,1X,D15.8)
 121                CONTINUE
                  CLOSE(UNIT=30)
C     END IF
C
      STOP
      END
C=========================================================================
C
C
C.........................................................................
C FORCE:                                                                 .
C  Calcul de la Force en Utilisant l'Algorithme 'The Short-Range Force ' .
C  Les énergies Inherentes aus système sont présentées par des potentiels 12 - 6
C  Leonnard - Jones .                                                    .
C.........................................................................
C
C===================  FORCE  =============================================
C
C
      SUBROUTINE FORCE
```

```
      IMPLICIT REAL*8(A-H,O-Z)
C Calcul de la force en utilisant l^@algorithme de 'THE SHORT-RANGE
C  FORCE'
C
      REAL*8 XX(200),YY(200),ZZ(200),AXX(200),AYY(200),AZZ(200)
     1,CXX(200),CYY(200)
      INTEGER HOC(50,50),NUM(50,50),LL(2601),C(4),B,D,O,Q,Y1
        COMMON/COOR/X(2601),Y(2601),Z(2601),CX(2601),CY(2601),PPX(2601),PP
     1Y(2601),VX(2601),VY(2601),VZ(2601),AX(2601),AY(2601),AZ(2601),NMOL,NMOL2,NMOL3
     COMMON/TEMPER/TEMFAC,TEMP,TK,DELT,AMASS,VDEL,DELT2,DELT3,NTEM,NTEM2,NTEM3
     COMMON/CELL/A1B2,A1B,A2B2,A2B,COXY,COYY,OCXY,OCYY,ALPHA
     COMMON/DISL/DIS(3,50),CDIS(3,50),NDIS
       COMMON/APPROX/U(132),V(132),W(132),DU(132),DV(132),DW(132),U0(1
     128),V0(128),W0(128),DU0(128),DV0(128),DW0(128)
     COMMON/DICH/E,E1,E2,SCAL3,SCAL4,NFLAG
      COMMON/PAR/SCAL2,PRES1,PRES2,EK,FM,FM2,PRESS,PRES,ATOT,AREA2,AREA0
     1,AREA,AREA1,AREA3,A,AREAP,AREAS,F,FM3
     COMMON/ITER/NS,NS1,NS2,NPAS,NTEST
     COMMON/RU/IRUN
     COMMON/PER/PMU,PLA,PMU0,PLA0,IDRA
      COMMON/ZUT/A1,SOM,SOU,TSOM,TSOU,GAMMA,LUT,LUT1,LUT2,LA,TEST,TTEST ,LB
C
      DATA RM/13.14D0/RG/13.14D0/
C
      E=0.D0
      IF(NS.EQ.1) F=0.D0
      PRES2=0.D0
      DO 1111 M=1,NMOL


         AX(M)=0.D0
           AY(M)=0.D0
C           AZ(M)=0.D0
 1111  CONTINUE
      NIBOIT=DINT(A1B2/RG)
      NJBOIT=DINT(A2B2/RG)
      HCX=A1B2/DFLOAT(NIBOIT)
      HCY=A2B2/DFLOAT(NJBOIT)
      DO 1 I=1,NIBOIT
         DO 1 J=1,NJBOIT
            HOC(I,J)=0
            NUM(I,J)=0
1     CONTINUE
C
      DO 2 M=1,NMOL
         DO 3 N=1,NDIS
            IF(DFLOAT(M).EQ.DIS(3,N)) THEN
                           GOTO 2
            ENDIF
 3       CONTINUE
         I=DINT((CX(M)+A1B)/HCX)+1
         J=DINT((CY(M)+A2B)/HCY)+1
         LL(M)=HOC(I,J)
         HOC(I,J)=M
         NUM(I,J)=NUM(I,J)+1
  2   CONTINUE
```

```
C
      DO 9 I=1,NIBOIT
         DO 9 J=1,NJBOIT
C           I=1
C           J=1
            IP=I
            JP=J
            Y1=HOC(IP,JP)
            NEND=NUM(IP,JP)
            NB=NEND
            DO 25  IMOL=1,NEND
               XX(IMOL)=X(Y1)
               YY(IMOL)=Y(Y1)
C              ZZ(IMOL)=Z(Y1)
               AXX(IMOL)=AX(Y1)
               AYY(IMOL)=AY(Y1)
C              AZZ(IMOL)=AZ(Y1)
C               WRITE(3,*) Y1,IP,JP
               Y1=LL(Y1)
25          CONTINUE
           IMOL=NEND
            NSTART=IMOL+1
C           NB=NSTART-1
CCC          JP=J   FROM BEFORE
            IF(I.GT.1) THEN
                  XSHIFT=0.0
                  IP=I-1
               ELSE
                  XSHIFT=-A1B2
                  IP=NIBOIT


            ENDIF
            Y1=HOC(IP,JP)
            NEND=NUM(IP,JP)+IMOL
            DO 28 IMOL=NSTART,NEND
               XX(IMOL)=X(Y1)+XSHIFT
               YY(IMOL)=Y(Y1)
C              ZZ(IMOL)=Z(Y1)
               AXX(IMOL)=AX(Y1)
               AYY(IMOL)=AY(Y1)
C              AZZ(IMOL)=AZ(Y1)
C               WRITE(3,*) Y1,IP,JP
               Y1=LL(Y1)
 28          CONTINUE
            IMOL=NEND
            NSTART=IMOL+1
CCC          IP=I-1 OR NIBOIT AS SET BEFORE
            IF(J.GT.1)  THEN
                  JP=J-1
                  YSHIFT=0.0
               ELSE
                  XSHIFT=-A2B2*COXY+XSHIFT
                  YSHIFT=-A2B2*COYY
                  JP=NJBOIT
            ENDIF
```

```
      Y1=HOC(IP,JP)
      NEND=NUM(IP,JP)+IMOL
      DO 31 IMOL=NSTART,NEND
         XX(IMOL)=X(Y1)+XSHIFT
         YY(IMOL)=Y(Y1)+YSHIFT
C         ZZ(IMOL)=Z(Y1)
         AXX(IMOL)=AX(Y1)
         AYY(IMOL)=AY(Y1)
C         AZZ(IMOL)=AZ(Y1)
C           WRITE(3,*) Y1,IP,JP
         Y1=LL(Y1)
 31     CONTINUE
      IMOL=NEND
      NSTART=IMOL+1
      IP=I
      IF(I.EQ.1)  THEN
                 XSHIFT=A1B2+XSHIFT
      ENDIF
      Y1=HOC(IP,JP)
      NEND=NUM(IP,JP)+IMOL
      DO 34 IMOL=NSTART,NEND
         XX(IMOL)=X(Y1)+XSHIFT
         YY(IMOL)=Y(Y1)+YSHIFT
C         ZZ(IMOL)=Z(Y1)
         AXX(IMOL)=AX(Y1)
         AYY(IMOL)=AY(Y1)
C         AZZ(IMOL)=AZ(Y1)
C           WRITE(3,*) Y1,IP,JP
         Y1=LL(Y1)
 34     CONTINUE
      IMOL=NEND
      NSTART=IMOL+1


CCC        JP=J-1 OR NJBOIT AS FROM BEFORE
      IF(I.LT.NIBOIT) THEN
                   IP=I+1
                ELSE
                   XSHIFT=A1B2+XSHIFT
                   IP=1
      ENDIF
      Y1=HOC(IP,JP)
      NEND=NUM(IP,JP)+IMOL
      DO 36 IMOL=NSTART,NEND
         XX(IMOL)=X(Y1)+XSHIFT
         YY(IMOL)=Y(Y1)+YSHIFT
C         ZZ(IMOL)=Z(Y1)
         AXX(IMOL)=AX(Y1)
         AYY(IMOL)=AY(Y1)
C         AZZ(IMOL)=AZ(Y1)
C           WRITE(3,*) Y1,IP,JP
         Y1=LL(Y1)
 36     CONTINUE
C       NEND=IMOL
      DO 50 D=1,NB
         XK=XX(D)
```

```
               YK=YY(D)
C               ZK=ZZ(D)
               AXK=AXX(D)
               AYK=AYY(D)
C               AZK=AZZ(D)
               DO 40 B=D+1,NEND
                    CMPX=XK-XX(B)
                    CMPY=YK-YY(B)
C                    CMPZ=ZK-ZZ(B)
                     R2=CMPX**2+CMPY**2
C                      R3=CMPX**2+CMPY**2+CMPZ**2
                    IF(R2.LE.172.6596D0) THEN

                                   R=DSQRT(R2)
                                    DK=R*10.0
                                   K=DINT(DK)
                                    E=E+U(K)+V(K)*R+W(K
    1                               )*R2
                                   DE=DU(K)+DV(K)*R+DW
    1                               (K)*R2
                                   V1=DE*TK
                                    AXX(B)=AXX(B)-V1*CM
    1                              PX
                                    AXK   =AXK   +V1*CM
    1                              PX
                                    AYY(B)=AYY(B)-V1*CM
    1                              PY
                                    AYK   =AYK   +V1*CM
    1                              PY
C                                   AZZ(B)=AZZ(B)-V1*CM
C   1                              PZ
C                                  AZK   =AZK   +V1*CM
C   1                              PZ
                                   PRES2=PRES2+V1*R2*1

    1                              .D16
C   39                          CONTINUE
                    ENDIF
40              CONTINUE
              AXX(D)=AXK
              AYY(D)=AYK
C              AZZ(D)=AZK
50           CONTINUE
           IP=I
           JP=J
           Y1=HOC(IP,JP)
           NEND=NUM(IP,JP)
           DO 55  IMOL=1,NEND
             AX(Y1)=AXX(IMOL)
             AY(Y1)=AYY(IMOL)
C             AZ(Y1)=AZZ(IMOL)
             Y1=LL(Y1)
55           CONTINUE
           IMOL=NEND
           NSTART=IMOL+1
```

```
CCC        JP=J   FROM BEFORE
          IF(I.GT.1) THEN
                    IP=I-1
               ELSE
                    IP=NIBOIT
               ENDIF
          Y1=HOC(IP,JP)
          NEND=NUM(IP,JP)+IMOL
          DO 58 IMOL=NSTART,NEND
            AX(Y1)=AXX(IMOL)
            AY(Y1)=AYY(IMOL)
C            AZ(Y1)=AZZ(IMOL)
            Y1=LL(Y1)
58          CONTINUE
          IMOL=NEND
          NSTART=IMOL+1
CCC        IP=I-1 OR NIBOIT AS SET BEFORE
          IF(J.GT.1)  THEN
                     JP=J-1
               ELSE
                     JP=NJBOIT
               ENDIF
          Y1=HOC(IP,JP)
          NEND=NUM(IP,JP)+IMOL
          DO 61 IMOL=NSTART,NEND
            AX(Y1)=AXX(IMOL)
            AY(Y1)=AYY(IMOL)
C            AZ(Y1)=AZZ(IMOL)
            Y1=LL(Y1)
61          CONTINUE
          IMOL=NEND
          NSTART=IMOL+1
          IP=I
          Y1=HOC(IP,JP)
          NEND=NUM(IP,JP)+IMOL
          DO 64 IMOL=NSTART,NEND


           AX(Y1)=AXX(IMOL)
            AY(Y1)=AYY(IMOL)
C            AZ(Y1)=AZZ(IMOL)
            Y1=LL(Y1)
64          CONTINUE
          IMOL=NEND
          NSTART=IMOL+1
CCC        JP=J-1 OF NJBOIT AS FROM BEFORE
          IF(I.LT.NIBOIT) THEN
                        IP=I+1
                  ELSE
                        IP=1
                  ENDIF
          Y1=HOC(IP,JP)
          NEND=NUM(IP,JP)+IMOL
          DO 66 IMOL=NSTART,NEND
            AX(Y1)=AXX(IMOL)
            AY(Y1)=AYY(IMOL)
```

```
C              AZ(Y1)=AZZ(IMOL)
             Y1=LL(Y1)
66           CONTINUE
           NEND=IMOL
 9    CONTINUE
      IF(NTEST.EQ.1.AND.NDEP.EQ.0) THEN
                F=F+E
                 WRITE (11,*) NS,E/DFLOAT(NMOL3)
   1             ,F/DFLOAT(NMOL3)/DFLOAT(NS-NTEM3-1)
                ELSE
                  WRITE(11,*) NS,E/DFLOAT(NMOL3),0.0000
      ENDIF
C
      RETURN
      END
C=============================================================================
C
C
C.............................................................................
C FORCE EN Z :                                                                .
C                                                                             .
C  La troisième dimension est due à la présence  du  Substrat .               .
C                                                                             .
C.............................................................................
C
C====================== FORCE EN Z ==========================================
C
C
      SUBROUTINE FZ
       IMPLICIT REAL*8(A-H,O-Z)
C
        COMMON/COOR/X(2601),Y(2601),Z(2601),CX(2601),CY(2601),PPX(2601),PP
        1Y(2601),VX(2601),VY(2601),VZ(2601),AX(2601),AY(2601),AZ(2601),NMOL
   1,NMOL2,NMOL3
    COMMON/TEMPER/TEMFAC,TEMP,TK,DELT,AMASS,VDEL,DELT2,DELT3,NTEM,NTEM
   12,NTEM3
     COMMON/CELL/A1B2,A1B,A2B2,A2B,COXY,COYY,OCXY,OCYY,ALPHA
     COMMON/DISL/DIS(3,50),CDIS(3,50),NDIS


   COMMON/APPROX/U(132),V(132),W(132),DU(132),DV(132),DW(132),U0(1
    128),V0(128),W0(128),DU0(128),DV0(128),DW0(128)
    COMMON/DICH/E,E1,E2,SCAL3,SCAL4,NFLAG
     COMMON/PAR/SCAL2,PRES1,PRES2,EK,FM,FM2,PRESS,PRES,ATOT,AREA2, AREA0
    1,AREA,AREA1,AREA3,A,AREAP,AREAS,F,FM3
    COMMON/ITER/NS,NS1,NS2,NPAS,NTEST
    COMMON/RU/IRUN
    COMMON/PER/PMU,PLA,PMU0,PLA0,IDRA
      COMMON/ZUT/A1,SOM,SOU,TSOM,TSOU,GAMMA,LUT,LUT1,LUT2,LA,TEST,TTEST
   1,LB
C
       DATA RM/13.14D0/RG/13.14D0/
C
      DO 1 M=1,NMOL
          DO 2 N=1,NDIS
              IF(DFLOAT(M).EQ.DIS(3,N)) THEN
```

```
C                               WRITE (3,*) 'DIS',M
                              ELSE
                              GOTO 2
         ENDIF
 2       CONTINUE
         DK=Z(M)*10.D0
         K=DINT(DK)
          E=E+U0(K)+V0(K)*Z(M)+W0(K)*(Z(M)**2)
          DE=DU0(K)+DV0(K)*Z(M)+DW0(K)*(Z(M)**2)
C        WRITE(3,*) DE
         AZ(M)=AZ(M)+DE*TK
 1       CONTINUE
         WRITE(11,555) EZ
 555      FORMAT(1X,'ENRGIE SELON Z=',1PE15.8)
C
C     WRITE (3,*) 'fz'
      RETURN
      END
C-----------------------------------------------------------------------
C
C
C************************************************************************
C MOUVEMENT  A VOLUME CONSTANT :                                        *
C Dynamique moleculaire ensemble canonique (N,V,T)  méthode
C des forces amorties.                                                  *
C************************************************************************
C
C
C--------------  MOUVEMENT A VOLUME CONSTANT  ---------------------------
C
C
      SUBROUTINE MOUVE2
     IMPLICIT REAL*8(A-H,O-Z)
        COMMON/UL/VPX(2601),VPY(2601),VPZ(2601),P1X(2601),P1Y(2601),P2X(26
      101),P2Y(2601),PX(2601),PY(2601),TRX(2601),TRY(2601),XP(2601),
     1YP(2601),P(100)
C
        COMMON/COOR/X(2601),Y(2601),Z(2601),CX(2601),CY(2601),PPX(2601),PP
      1Y(2601),VX(2601),VY(2601),VZ(2601),AX(2601),AY(2601),AZ(2601),NMOL,NMOL2,NMOL3
     COMMON/TEMPER/TEMFAC,TEMP,TK,DELT,AMASS,VDEL,DELT2,DELT3,NTEM,NTEM2,NTEM3


     COMMON/CELL/A1B2,A1B,A2B2,A2B,COXY,COYY,OCXY,OCYY,ALPHA
     COMMON/DISL/DIS(3,50),CDIS(3,50),NDIS
     COMMON/PAR/SCAL2,PRES1,PRES2,EK,FM,FM2,PRESS,PRES,ATOT,AREA2, AREA0
    1,AREA,AREA1,AREA3,A,AREAP,AREAS,F,FM3
     COMMON/ITER/NS,NS1,NS2,NPAS,NTEST
     COMMON/DICH/E,E1,E2,SCAL3,SCAL4,NFLAG
     COMMON/PER/PMU,PLA,PMU0,PLA0,IDRA
     COMMON/ZUT/A1,SOM,SOU,TSOM,TSOU,GAMMA,LUT,LUT1,LUT2,LA,TEST,TTEST ,LB
     COMMON/PR/PRES0,BETA,TMX,TMY,TVX,TVY
     COMMON/RU/IRUN
     COMMON/POSMOY/XMOY(2601),YMOY(2601)
C    COMMON/BOUCLE/ZI,ZMOY
C
C 2 dimensions=1.d0       3 dimensions=3.d0/2.d0
```

```
      TEMFA1=TEMP*(TK*1.D20)
      TEMFA2=.5D0*TEMP*(TK*1.D20)
      R3=DSQRT(3.D0)
      R0=1000.D0
      IF(NS.EQ.1) THEN
            AREA0=(A1B2*A2B2)*R3/2.D0
            A=0.D0
            PRESS=0.D0
            ATOT=0.D0
            SCAL2=1.D0
            AREA=AREA0
      ENDIF
       IF(NS.EQ.NTEM.AND.DELT3.NE.DELT) THEN
                  CALL DISVIT
                  BETA=DELT3/DELT
                  PDELT=DELT
                  DELT=DELT3
                  VDEL=1.D8*DELT
                   DELT2=(DELT*1.D28)/AMASS
                  FM=FM*BETA
                  FM2=FM2*BETA
                  FM3=FM
          ENDIF
      PRES1=0.D0
      PRES=0.D0
       EM=FM3*AMASS*1.D-4/(3.92D0)**2
      WRITE(12,*) 'EM=',EM
C      ALPHA=1.D14
.C
      IF(NS.LE.(NTEM+1).AND.NS.GE.NTEM) THEN
            GOTO 101
      ENDIF
C
      TMX=TMX/DFLOAT(NMOL3)
      TMY=TMY/DFLOAT(NMOL3)
      TVX=TVX/DFLOAT(NMOL3)
      TVY=TVY/DFLOAT(NMOL3)
      PXT=0.0
      PYT=0.0
      DO 100 M=1,NMOL
         IF(NDIS.NE.0) THEN
                  DO 7 N=1,NDIS


                     IF(DFLOAT(M).EQ.DIS(3,N)) GOTO 100
 7                CONTINUE
C                 IF(IDRA.NE.0) GOTO 3
C                 IF(M.LE.LA) GOTO 1
C                 IF(M.GT.(NMOL-LA)) GOTO 1
C                 IF(MOD(M,LA).EQ.0) GOTO 1
C                 IF(MOD(M,LA).EQ.1) GOTO 1
         ENDIF
         P2X(M)=P1X(M)
          P1X(M)=(VX(M)-TVX+TRX(M)-TMX)/2.D0
         TRX(M)=VX(M)-TVX
         P2Y(M)=P1Y(M)
```

```
          P1Y(M)=(VY(M)-TVY+TRY(M)-TMY)/2.D0
         TRY(M)=VY(M)-TVY
          PX(M)=2.D0*P1X(M)-P2X(M)
          PY(M)=2.D0*P1Y(M)-P2Y(M)
         PXT=PX(M)+PXT
         PYT=PY(M)+PYT
 100  CONTINUE
    PXT=PXT/DFLOAT(NMOL3)
    PYT=PYT/DFLOAT(NMOL3)
    DO 199 M=1,NMOL
         IF(NDIS.NE.0) THEN
                     DO 17 N=1,NDIS
                        IF(DFLOAT(M).EQ.DIS(3,N)) GOTO 199
 17                   CONTINUE
C                     IF(IDRA.NE.0) GOTO 3
C                     IF(M.LE.LA) GOTO 1
C                     IF(M.GT.(NMOL-LA)) GOTO 1
C                     IF(MOD(M,LA).EQ.0) GOTO 1
C                     IF(MOD(M,LA).EQ.1) GOTO 1
         ENDIF
         PX(M)=PX(M)-PXT
         PY(M)=PY(M)-PYT
C         TRX(M)=TRX(M)-TMX
C         TRY(M)=TRY(M)-TMY
            PRES1=PRES1+AMASS*((PX(M))**2+(PY(M))**2)*1.D-4
 199  CONTINUE
    IF(NS.LE.2) THEN
            PRES1=TEMFAC*2.D-4*DFLOAT(NMOL3)
    ENDIF
 101  PRES=(PRES1+PRES2)/2.D0/AREA
C
    TVX=0.D0
    TVY=0.D0
    TMX=0.D0
    TMY=0.D0
C
C execs de demo
    IF(NTEST.EQ.0) THEN
CC             IF(NS.GT.NTEM2) THEN
            IF(NS.GT.NTEM2.AND.DABS(PRES-PRES0).LT..1D0) THEN
                                    FM3=FM2
                                    NTEST=1
                                   NTEM3=NS-1
            ENDIF


    ENDIF
    IF (NTEST.EQ.1) THEN
            PRESS=PRESS+PRES
C             ATOT=ATOT+AREA2
             A=A+A1
             WRITE(7,*) NS,PRES,PRESS/DFLOAT(NS-NTEM3)
              WRITE(4,*) NS,A1,A/DFLOAT(NS-NTEM3)
             ELSE
           WRITE(7,*) NS,PRES
           WRITE(4,*) NS,A1
```

```
      ENDIF
      IF(NS.EQ.NPAS+IRUN) THEN
                WRITE(7,*) 'pres1=',PRES1,PRES1/2.D0/AREA
                WRITE(7,*) 'pres2=',PRES2,PRES2/2.D0/AREA
               WRITE(7,*) 'pres=',PRES
                WRITE(7,*) 'pression=',PRESS/DFLOAT(NS-NTEM3)
               WRITE(7,*) 'cot-=',A1B2
                WRITE(7,*) 'parametre=',A1B2/DFLOAT(LA)
                WRITE(7,*) 'parametrem=',A/DFLOAT(NS-NTEM3)
               WRITE(7,*) 'surfC=',AREA0
                WRITE(7,*) 'enem=',F/DFLOAT(NMOL3)/DFLOAT(NS-NTEM3- 1)
               WRITE(7,*) 'delt=',DELT
               WRITE(7,*) 'temp=',TEMP
                 MO=DINT(1.D-17/DELT/(DSQRT(3*TK*TEMP/AMASS)))
               WRITE(7,*) 'mo2=',MO
      ENDIF
C
      EK=0.D0
C
       DO 1 M=1,NMOL
          IF(NDIS.NE.0) THEN
                      DO 2 N=1,NDIS
                         IF(DFLOAT(M).EQ.DIS(3,N)) GOTO 1
 2                    CONTINUE
C                      IF(IDRA.NE.0) GOTO 3
C                      IF(M.LE.LA) GOTO 1
C                      IF(M.GT.(NMOL-LA)) GOTO 1
C                      IF(MOD(M,LA).EQ.0) GOTO 1
C                      IF(MOD(M,LA).EQ.1) GOTO 1
 3        ENDIF
C         ENDIF
            VPX(M)= TRX(M)+(AX(M)*1.D28/AMASS)*DELT/2.D0
          PPX(M)=VPX(M)
            VPY(M)= TRY(M)+(AY(M)*1.D28/AMASS)*DELT/2.D0
          PPY(M)=VPY(M)
C          VPZ(M)=VZ(M)+AZ(M)*DELT2/2.D0
          EK=EK+PPX(M)**2+PPY(M)**2
C          EK1=EK1+VPZ(M)**2
 1     CONTINUE
       EK=EK*AMASS/DFLOAT(NMOL3-1)*.5D0
C       EK1=EK1*AMASS/DFLOAT(NMOL3)*.5D0
      SCAL2=DSQRT(TEMFAC/EK)
C       SCAL5=DSQRT(TEMFA2/EK1)
      WRITE(5,*) NS,SCAL2
C
      DX=A1B2


      DY=A2B2
C      A1B3=DSQRT(2.D0*AREA*(1.0-DFLOAT(NDIS)/DFLOAT(NMOL))/R3)
      A1B2=DSQRT(2.D0*AREA*ALPHA/R3)
      A2B2=DSQRT(2.D0*AREA/ALPHA/R3)
      A1=A1B2/DFLOAT(LA)
      A1B=A1B2/2.D0
      A2B=A2B2/2.D0
C
```

```
      DO 4 M=1,NMOL
        IF(NDIS.NE.0) THEN
                DO 5 N=1,NDIS
                  IF(DFLOAT(M).EQ.DIS(3,N)) GOTO 4
 5              CONTINUE
C                IF(IDRA.NE.0) GOTO 6
C                IF(M.LE.LA) GOTO 4
C                IF(M.GT.(NMOL-LA)) GOTO 4
C                IF(MOD(M,LA).EQ.0) GOTO 4
C                IF(MOD(M,LA).EQ.1) GOTO 4
 6        ENDIF
          VX(M)=TRX(M)*(2.D0*SCAL2-1.D0)+SCAL2*(AX
     1    (M)*1.D28/AMASS)*DELT
        XP(M)=X(M)
        X(M)=X(M)      +VDEL*VX(M)
          VY(M)=TRY(M)*(2.D0*SCAL2-1.D0)+SCAL2*(AY
     1    (M)*1.D28/AMASS)*DELT
        YP(M)=Y(M)
        Y(M)=Y(M)      +VDEL*VY(M)
C
C          VZ(M)=VZ(M)*(2.D0*SCAL5-1.D0)+AZ(M)*DELT2*SCAL5
C        Z(M)=Z(M)      +VDEL*VZ(M)
C
C        ZTOT=ZTOT+Z(M)
C
C  Les coordonnées sont transformées au système crystallographique
C
        CX(M)=X(M)+OCXY*Y(M)
        CY(M)=OCYY*Y(M)
C Les conditions périodiques sont testées .
C
        IF(CX(M).GT.A1B) THEN
                CX(M)=CX(M)-A1B2
                X(M)=X(M)-A1B2
                VX(M)=VX(M)
        ELSE IF(CX(M).LE.-A1B) THEN
                  CX(M)=CX(M)+A1B2
                  X(M)=X(M)+A1B2
                  VX(M)=VX(M)
        END IF
        IF(CY(M).GT.A2B) THEN
                CY(M)=CY(M)-A2B2
                X(M)=X(M)-COXY*A2B2
                Y(M)=Y(M)-COYY*A2B2
                VX(M)=VX(M)
                VY(M)=VY(M)
        ELSE IF(CY(M).LE.-A2B) THEN
                  CY(M)=CY(M)+A2B2

                X(M)=X(M)+COXY*A2B2
                  Y(M)=Y(M)+COYY*A2B2
                  VX(M)=VX(M)
                  VY(M)=VY(M)
        END IF
C        IF(Z(M).GT.9.D0) THEN
```

```
C                    VZ(M)=-VZ(M)
C         ENDIF
          TMX=TMX+TRX(M)
          TVX=TVX+VX(M)
          TMY=TMY+TRY(M)
          TVY=TVY+VY(M)
 4     CONTINUE
C      ZMOY=ZMOY+ZTOT
C       WRITE(9,*) NS,ZTOT/DFLOAT(NMOL3),ZMOY/DFLOAT(NMOL3)/DFLOAT(NS)
C      ZTOT=0.D0
       IF(NDEP.EQ.0.AND.NS.EQ.5) THEN
        DO 23 M=1,NMOL
          WRITE(15,*) X(M),Y(M)
 23      CONTINUE
         ENDIF
C      WRITE (3,*) 'mouve2'
      RETURN
      END
C===========================================================================
C
C
C...........................................................................
C MOUVEMENT A PRESSION ET TEMPÈRATURE CONSTANTES :                          .
C  Introduction de la dynamique moléculaire ,ensemble canonique (N,P,T)     .
C  méthode combinée d' Anderson .                                           .
C...........................................................................
C
C
C=========MOUVEMENT A PRESSION ET TEMPERATURE CONSTANTE =====================
C
C
      SUBROUTINE MOUVE3
     IMPLICIT REAL*8(A-H,O-Z)
        COMMON/UL/VPX(2601),VPY(2601),VPZ(2601),P1X(2601),P1Y(2601),P2X(26
             101),P2Y(2601),PX(2601),PY(2601),TRX(2601),TRY(2601),XP(2601),YP(2601),P(100)

     COMMON/TEMPER/TEMFAC,TEMP,TK,DELT,AMASS,VDEL,DELT2,DELT3,NTEM,NTEM2,NTEM3,
     COMMON/CELL/A1B2,A1B,A2B2,A2B,COXY,COYY,OCXY,OCYY,ALPHA
     COMMON/DISL/DIS(3,50),CDIS(3,50),NDIS
     COMMON/PAR/SCAL2,PRES1,PRES2,EK,FM,FM2,PRESS,PRES,ATOT,AREA2,AREA0
    1,AREA,AREA1,AREA3,A,AREAP,AREAS,F,FM3
     COMMON/ITER/NS,NS1,NS2,NPAS,NTEST
     COMMON/DICH/E,E1,E2,SCAL3,SCAL4,NFLAG
     COMMON/PER/PMU,PLA,PMU0,PLA0,IDRA
      COMMON/ZUT/A1,SOM,SOU,TSOM,TSOU,GAMMA,LUT,LUT1,LUT2,LA,TEST,TTEST ,LB
     COMMON/PR/PRES0,BETA,TMX,TMY,TVX,TVY
     COMMON/RU/IRUN
     COMMON/POSMOY/XMOY(2601),YMOY(2601)
C    COMMON/BOUCLE/ZI,ZMOY
C

C 2 dimensions=1.d0      3 dimensions=3.d0/2.d0
     TEMFA1=TEMP*(TK*1.D20)
     TEMFA2=.5D0*TEMP*(TK*1.D20)
     R3=DSQRT(3.D0)
```

```
      R0=1000.D0
      IF(NS.EQ.1) THEN
            AREA0=(A1B2*A2B2)*R3/2.D0
           A=0.D0
           PRESS=0.D0
           ATOT=0.D0
           SCAL2=1.D0
           AREA=AREA0
           AREA1=AREA0
           AREA2=AREA0
           DO 10 M=1,NMOL
                XMOY(M)=0.D0
 10              YMOY(M)=0.D0
      ENDIF
       IF(NS.EQ.NTE*1.AND.DELT3.NE.DELT) THEN
                   CALL DISVIT
                   BETA=DELT3/DELT
                   PDELT=DELT
                   DELT=DELT3
                   VDEL=1.D8*DELT
                    DELT2=(DELT*1.D28)/AMASS
                   FM=FM*BETA
                   FM2=FM2*BETA
                   FM3=FM
                   AREA3=AREA
                   AREA2=AREA
                   AREA1=AREA
          ENDIF
      PRES1=0.D0
      PRES=0.D0
       EM=FM3*AMASS*1.D-4/(3.92D0)**2
      WRITE(12,*) 'EM=',EM
C     ALPHA=1.D14
      IF(NS.LE.(NTEM+1).AND.NS.GE.NTEM) THEN
          GOTO 101
      ENDIF
      TMX=TMX/DFLOAT(NMOL3)
      TMY=TMY/DFLOAT(NMOL3)
      TVX=TVX/DFLOAT(NMOL3)
      TVY=TVY/DFLOAT(NMOL3)
      PXT=0.0
      PYT=0.0
C         WRITE(3,*) 'IDRA=',IDRA
      DO 100 M=1,NMOL
          IF(NDIS.NE.0) THEN
                   DO 7 N=1,NDIS
                      IF(DFLOAT(M).EQ.DIS(3,N)) GOTO 100
 7                 CONTINUE
C                   IF(IDRA.NE.0) GOTO 3
C                   IF(M.LE.LA) GOTO 1
C                   IF(M.GT.(NMOL-LA)) GOTO 1
C                   IF(MOD(M,LA).EQ.0) GOTO 1

C                   IF(MOD(M,LA).EQ.1) GOTO 1
          ENDIF
```

```
         P2X(M)=P1X(M)
          P1X(M)=(VX(M)-TVX+TRX(M)-TMX)/2.D0-
     1                            XP(M)*AREAP*1.D-8/2.D0/AREA2
         TRX(M)=VX(M)-TVX
         P2Y(M)=P1Y(M)
          P1Y(M)=(VY(M)-TVY+TRY(M)-TMY)/2.D0-
     1                            YP(M)*AREAP*1.D-8/2.D0/AREA2
         TRY(M)=VY(M)-TVY
          PX(M)=2.D0*P1X(M)-P2X(M)
          PY(M)=2.D0*P1Y(M)-P2Y(M)
         PXT=PX(M)+PXT
         PYT=PY(M)+PYT
 100  CONTINUE
      PXT=PXT/DFLOAT(NMOL3)
      PYT=PYT/DFLOAT(NMOL3)
      DO 199 M=1,NMOL
          IF(NDIS.NE.0) THEN
                      DO 17 N=1,NDIS
                          IF(DFLOAT(M).EQ.DIS(3,N)) GOTO 199
 17                   CONTINUE
C                     IF(IDRA.NE.0) GOTO 3
C                     IF(M.LE.LA) GOTO 1
C                     IF(M.GT.(NMOL-LA)) GOTO 1
C                     IF(MOD(M,LA).EQ.0) GOTO 1
C                     IF(MOD(M,LA).EQ.1) GOTO 1
         ENDIF
         PX(M)=PX(M)-PXT
         PY(M)=PY(M)-PYT
            PRES1=PRES1+AMASS*((PX(M))**2+(PY(M))**2)*1.D-4
 199  CONTINUE
      IF(NS.LE.2) THEN
              PRES1=TEMFAC*2.D-4*DFLOAT(NMOL3)
      ENDIF
 101  PRES=(PRES1+PRES2)/2.D0/AREA
      AREAS=(PRES-PRES0)/EM
      AREA=2.D0*AREA1-AREA2+AREAS*DELT**2
      AREAP=(AREA-AREA2)/DELT/2.D0
      AREA3=AREA2
      AREA2=AREA1
      AREA1=AREA
C
      TVX=0.D0
      TVY=0.D0
      TMX=0.D0
      TMY=0.D0
C execs de demo
      IF(NTEST.EQ.0) THEN
CC              IF(NS.GT.NTEM2) THEN
            IF(NS.GT.NTEM2.AND.DABS(PRES-PRES0).LT..1D0) THEN
                                             FM3=FM2
                                             NTEST=1
                                            NTEM3=NS-1
            ENDIF
      ENDIF
```

```
 IF (NTEST.EQ.1) THEN
          PRESS=PRESS+PRES
          ATOT=ATOT+AREA2
          A=A+A1
          WRITE(7,*) NS,PRES,PRESS/DFLOAT(NS-NTEM3)
          WRITE(4,*) NS,A1,A/DFLOAT(NS-NTEM3)
          ELSE
         WRITE(7,*) NS,PRES
         WRITE(4,*) NS,A1
   ENDIF
   IF(NS.EQ.NPAS+IRUN) THEN
          WRITE(7,*) 'pres1=',PRES1,PRES1/2.D0/AREA
          WRITE(7,*) 'pres2=',PRES2,PRES2/2.D0/AREA
          WRITE(7,*) 'pres=',PRES
          WRITE(7,*) 'pression=',PRESS/DFLOAT(NS-NTEM3)
          WRITE(7,*) 'cot-=',A1B2
          WRITE(7,*) 'parametre=',A1B2/DFLOAT(LA)
          WRITE(7,*) 'parametrem=',A/DFLOAT(NS-NTEM3)
          WRITE(7,*) 'surf0=',AREA0
          WRITE(7,*) 'surf=',AREA2
          WRITE(7,*) 'surfm=',ATOT/DFLOAT(NS-NTEM3)
          WRITE(7,*) 'apri=',AREAP
          WRITE(7,*) 'asec=',AREAS
          WRITE(7,*) 'enem=',F/DFLOAT(NMOL3)/DFLOAT(NS-NTEM3- 1)
          WRITE(7,*) 'delt=',DELT
          WRITE(7,*) 'temp=',TEMP
            MO=DINT(1.D-17/DELT/(DSQRT(3*TK*TEMP/AMASS)))
          WRITE(7,*) 'mo2=',MO
   ENDIF
   EK=0.D0
C
    DO 1 M=1,NMOL
      IF(NDIS.NE.0) THEN
              DO 2 N=1,NDIS
                IF(DFLOAT(M).EQ.DIS(3,N)) GOTO 1
 2            CONTINUE
C              IF(IDRA.NE.0) GOTO 3
C              IF(M.LE.LA) GOTO 1
C              IF(M.GT.(NMOL-LA)) GOTO 1
C              IF(MOD(M,LA).EQ.0) GOTO 1
C              IF(MOD(M,LA).EQ.1) GOTO 1
 3      ENDIF
          VPX(M)= TRX(M)+(AX(M)*1.D28/AMASS+X(M)/2.D0/AREA2*(AREAS-AR
   1     AP**2/2.D0/AREA2)*1.D-8)*DELT/2.D0
        PPX(M)=VPX(M)-X(M)*AREAP*1.D-8/2.D0/AREA2
          VPY(M)= TRY(M)+(AY(M)*1.D28/AMASS+Y(M)/2.D0/AREA2*(AREAS-AR
   1     AP**2/2.D0/AREA2)*1.D-8)*DELT/2.D0
        PPY(M)=VPY(M)-Y(M)*AREAP*1.D-8/2.D0/AREA2
C       VPZ(M)=VZ(M)+AZ(M)*DELT2/2.D0
        EK=EK+PPX(M)**2+PPY(M)**2
C       EK1=EK1+VPZ(M)**2
 1   CONTINUE
     EK=EK*AMASS/DFLOAT(NMOL3-1)*.5D0
C      WRITE(3,*)'EK=',EK
C    EK1=EK1*AMASS/DFLOAT(NMOL3)*.5D0
    SCAL2=DSQRT(TEMFAC/EK)
```

```
C     SCAL5=DSQRT(TEMFA2/EK1)
      WRITE(5,*) NS,SCAL2
C
     DX=A1B2
     DY=A2B2
C     A1B3=DSQRT(2.D0*AREA*(1.0-DFLOAT(NDIS)/DFLOAT(NMOL))/R3)
      A1B2=DSQRT(2.D0*AREA*ALPHA/R3)
      A2B2=DSQRT(2.D0*AREA/ALPHA/R3)
      A1=A1B2/DFLOAT(LA)
      A1B=A1B2/2.D0
      A2B=A2B2/2.D0
C
      DO 4 M=1,NMOL
         IF(NDIS.NE.0) THEN
                    DO 5 N=1,NDIS
                       IF(DFLOAT(M).EQ.DIS(3,N)) GOTO 4
 5                  CONTINUE
C                    IF(IDRA.NE.0) GOTO 6
C                    IF(M.LE.LA) GOTO 4
C                    IF(M.GT.(NMOL-LA)) GOTO 4
C                    IF(MOD(M,LA).EQ.0) GOTO 4
C                    IF(MOD(M,LA).EQ.1) GOTO 4
 6        ENDIF
            VX(M)=TRX(M)*(2.D0*SCAL2-1.D0)+X(M)*(1.D0-SCAL2)*AREAP*1.D-
     1     8/AREA2+SCAL2*(AX
     1         (M)*1.D28/AMASS+(X(M)/2.D0/AREA2)*(AREAS-AREAP**2/2.D0/AREA
     1     2)*1.D-8)*DELT
          XP(M)=X(M)
          X(M)=X(M)       +VDEL*VX(M)
            VY(M)=TRY(M)*(2.D0*SCAL2-1.D0)+Y(M)*(1.D0-SCAL2)*AREAP*1.D-
     1     8/AREA2+SCAL2*(AY
     1         (M)*1.D28/AMASS+(Y(M)/2.D0/AREA2)*(AREAS-AREAP**2/2.D0/AREA
     1     2)*1.D-8)*DELT
          YP(M)=Y(M)
          Y(M)=Y(M)       +VDEL*VY(M)
C          VZ(M)=VZ(M)*(2.D0*SCAL5-1.D0)+AZ(M)*DELT2*SCAL5
C         Z(M)=Z(M)       +VDEL*VZ(M)
C         ZTOT=ZTOT+Z(M)
C Les coordonnées sont  transformées au systéme  crystallographique .
C
          CX(M)=X(M)+OCXY*Y(M)
          CY(M)=OCYY*Y(M)
C Les conditions périodiques sont testées .
C
          IF(CX(M).GT.A1B) THEN
                    CX(M)=CX(M)-A1B2
                    X(M)=X(M)-A1B2
                       VX(M)=VX(M)-A1B2*AREAP/2.D0/AREA2*1.D-8
          ELSE IF(CX(M).LE.-A1B) THEN
                       CX(M)=CX(M)+A1B2
                       X(M)=X(M)+A1B2
                          VX(M)=VX(M)+A1B2*AREAP/2.D0/AREA2*1.
     1     D-8
          END IF
```

```
        IF(CY(M).GT.A2B) THEN
                 CY(M)=CY(M)-A2B2


                X(M)=X(M)-COXY*A2B2
                  Y(M)=Y(M)-COYY*A2B2
                     VX(M)=VX(M)-A2B2*COXY*AREAP/2.D0/AREA2*1.D
     1          -8
                     VY(M)=VY(M)-A2B2*COYY*AREAP/2.D0/AREA2*1.D
     1          -8
        ELSE IF(CY(M).LE.-A2B) THEN
                     CY(M)=CY(M)+A2B2
                     X(M)=X(M)+COXY*A2B2
                     Y(M)=Y(M)+COYY*A2B2
                       VX(M)=VX(M)+A2B2*COXY*AREAP/2.D0/ARE
     1               A2*1.D-8
                       VY(M)=VY(M)+A2B2*COYY*AREAP/2.D0/ARE
     1               A2*1.D-8
        END IF
C        IF(Z(M).GT.9.D0) THEN
C                  VZ(M)=-VZ(M)
C        ENDIF
        TMX=TMX+TRX(M)
        TMY=TMY+TRY(M)
        TVY=TVY+VY(M)
 4    CONTINUE
C     ZMOY=ZMOY+ZTOT
C      WRITE(9,*) NS,ZTOT/DFLOAT(NMOL3),ZMOY/DFLOAT(NMOL3)/DFLOAT(NS)
C     ZTOT=0.D0
      IF(NDEP.EQ.0.AND.NS.EQ.5) THEN
       DO 23 M=1,NMOL
          WRITE(15,*) X(M),Y(M)
 23     CONTINUE
        ENDIF
C     WRITE (3,*) 'mouve2'
      RETURN
      END
C=====================================================================
C
C
C============= RENORMALISATION DE  L'ÉNERGIE ==========================
C
C
      SUBROUTINE RENEC
     IMPLICIT REAL*8(A-H,O-Z)
C
       COMMON/UL/VPX(2601),VPY(2601),VPZ(2601),P1X(2601),P1Y(2601),P2X(26
     101),P2Y(2601),PX(2601),PY(2601),TRX(2601),TRY(2601),XP(2601),  YP(2601),P(100)
       COMMON/COOR/X(2601),Y(2601),Z(2601),CX(2601),CY(2601),PPX(2601),PP
     1Y(2601),VX(2601),VY(2601),VZ(2601),AX(2601),AY(2601),AZ(2601),NMOL,NMOL2,NMOL3
     COMMON/TEMPER/TEMFAC,TEMP,TK,DELT,AMASS,VDEL,DELT2,DELT3,NTEM,NTEM2,NTEM3
     COMMON/CELL/A1B2,A1B,A2B2,A2B,COXY,COYY,OCXY,OCYY,ALPHA
     COMMON/DISL/DIS(3,50),CDIS(3,50),NDIS
     COMMON/PER/PMU,PLA,PMU0,PLA0,IDRA
      COMMON/ZUT/A1,SOM,SOU,TSOM,TSOU,GAMMA,LUT,LUT1,LUT2,LA,TEST,TTEST,LB
C
```

```
      DIMENSION VB1(2),DMOL(2)
C

C Ne pas oublier vz


      DO 3 J=1,2
C        VB1(J)=0.D0
        DMOL(J)=0.D0
3     CONTINUE
C
      DO 1 M=1,NMOL
         IF(NDIS.NE.0) THEN
                     DO 2 N=1,NDIS
                        IF(DFLOAT(M).EQ.DIS(3,N)) GOTO 1
 2                   CONTINUE
                     IF(IDRA.NE.0) GOTO 6
                     IF(M.LE.LA) GOTO 1
                     IF(M.GT.(NMOL-LA)) GOTO 1
                     IF(MOD(M,LA).EQ.0) GOTO 1
                     IF(MOD(M,LA).EQ.1) GOTO 1
 6        ENDIF
         DMOL(1)=DMOL(1)+CX(M)
         DMOL(2)=DMOL(2)+CY(M)
 1    CONTINUE
C
C On  soustrait le vecteur vitesse de la boite .
C
C Ne pas oublier vz
      DO 4 J=1,2
        DMOL(J)=DMOL(J)/DFLOAT(NMOL3)
 4    CONTINUE
C
      DO 5 M=1,NMOL
         IF(NDIS.NE.0) THEN
                     DO 7 N=1,NDIS
                        IF(DFLOAT(M).EQ.DIS(3,N)) GOTO 5
 7                   CONTINUE
                     IF(IDRA.NE.0) GOTO 8
                     IF(M.LE.LA) GOTO 5
                     IF(M.GT.(NMOL-LA)) GOTO 5
                     IF(MOD(M,LA).EQ.0) GOTO 5
                     IF(MOD(M,LA).EQ.1) GOTO 5
 8        ENDIF
         CX(M)=CX(M)-DMOL(1)
         CY(M)=CY(M)-DMOL(2)
         X(M)=CX(M)+COXY*CY(M)
         Y(M)=     COYY*CY(M)
 5    CONTINUE

      WRITE(3,*)'TEMP=',TEMP
C      WRITE(3,*) 'renec'
      RETURN
      END
```

```
C------------------------------------------------------------------------
C
C**************************************************************************
C RECOMMENCE :                                                           *
C  Permet de continuer la procedure sans recommencer du début ,et suvegarder les données .   *
C**************************************************************************
C
C------------------------------- RECOMMENCE ------------------------------
C
C
      SUBROUTINE RECOM
       IMPLICIT REAL*8(A-H,O-Z)
C
      DIMENSION DMOL(2)
      REAL I,K
        COMMON/UL/VPX(2601),VPY(2601),VPZ(2601),P1X(2601),P1Y(2601),P2X(26
     101),P2Y(2601),PX(2601),PY(2601),TRX(2601),TRY(2601),XP(2601), YP(2601),P(100)
        COMMON/COOR/X(2601),Y(2601),Z(2601),CX(2601),CY(2601),PPX(2601),PP
     1Y(2601),VX(2601),VY(2601),VZ(2601),AX(2601),AY(2601),AZ(2601),NMOL,NMOL2,NMOL3
      COMMON/CELL/A1B2,A1B,A2B2,A2B,COXY,COYY,OCXY,OCYY,ALPHA
      COMMON/TEMPER/TEMFAC,TEMP,TK,DELT,AMASS,VDEL,DELT2,DELT3,NTEM,NTEM2,NTEM3
       COMMON/DISL/DIS(3,50),CDIS(3,50),NDIS
      COMMON/DICH/E,E1,E2,SCAL3,SCAL4,NFLAG
      COMMON/ITER/NS,NS1,NS2,NPAS,NTEST
       COMMON/ZUT/A1,SOM,SOU,TSOM,TSOU,GAMMA,LUT,LUT1,LUT2,LA,TEST,TTEST ,LB
       COMMON/PAR/SCAL2,PRES1,PRES2,EK,FM,FM2,PRESS,PRES,ATOT,AREA2,AREA0
     1,AREA,AREA1,AREA3,A,AREAP,AREAS,F,FM3
      COMMON/RU/IRUN
      COMMON/PER/PMU,PLA,PMU0,PLA0,IDRA
      COMMON/PR/PRES0,BETA,TMX,TMY,TVX,TVY
      COMMON/BOUCLE/ZI,ZMOY
C
C         WRITE(3,*) 'MVACrecom=',MVAC
      OPEN(UNIT=35)
      READ(35,*) MVAC
      READ(35,*) LA
      READ(35,*) LB
      READ(35,*) NMOL3
      READ(35,*) NMOL2
      READ(35,*) NDIS
      READ(35,*) A1
      READ(35,*) AREA
C      READ(35,*) AREA1
C      READ(35,*) AREA2
      READ(35,*) NTEM3
      READ(35,*) NTEST
      READ(35,*) F
C      READ(35,*) ATOT
```

```
      READ(35,*) PRESS
      READ(35,*) A
      READ(35,*) E1
      READ(35,*) E2
      READ(35,*) TMX
      READ(35,*) TMY
      READ(35,*) TVX
      READ(35,*) TVY


   IF(NTEST.EQ.1) THEN
              FM3=FM2
              ELSE
              FM3=FM
      ENDIF
      ALPHA=DFLOAT(LA)/DFLOAT(LB)
      A2=A1
C      Nombre de particules dans  la cellule unitaire .
      NMOLU=1
C     Nombre de cellules dans une rangée .
      LB=LA
C     Nombre de particules .
      NMOL=LA*LB*NMOLU
      L=0
          IF(NDIS.EQ.0) GOTO 111
          IF(NDIS.EQ.1) THEN
            DIS(3,L+1)=NMOL/2+1
            CDIS(3,L+1)=NMOL/2+1
          ELSE
       DO 18 N=NMOL/2-NDIS/2+1,NMOL/2+NDIS/2+1
          DIS(3,L+1)=DFLOAT(N)
          CDIS(3,L+1)=DFLOAT(N)
      L=L+1
18     CONTINUE
          ENDIF
111    DO 119 M=1,NMOL
          READ(35,7888) P1X(M),P1Y(M),P2X(M),P2Y(M)
7888        FORMAT(1X,D15.8,1X,D15.8,1X,D15.8,1X,D15.8)
119    CONTINUE
      DO 121 M=1,NMOL
          READ(35,7886) TRX(M),TRY(M),VX(M),VY(M)
7886        FORMAT(1X,D15.8,1X,D15.8,1X,D15.8,1X,D15.8)
121    CONTINUE
      CLOSE(UNIT=35)
      A1B2=DFLOAT(LA)*A1
      A1B=A1B2/2.D0
      A2B2=DFLOAT(LB)*A2
      A2B=A2B2/2.D0
C Quelques constantes utiles .
      CFDR=DATAN(1.D0)/45.D0
C
      COXY=DCOS(GAMMA*CFDR)
      COYY=DSIN(GAMMA*CFDR)
      OCXY=-COXY/COYY
      OCYY=1.D0/COYY
C
```

```
C       WRITE(3,*) 'TEST=',TEST
      IDRA=1
C Ces constantes sont utilisées pour changer du système crystallographique
C au système orthogonale et vice-versa.
      OPEN (UNIT=40)
      DO 88 M=1,NMOL
           READ(40,7880) X(M),Y(M)
           CX(M)=X(M)+OCXY*Y(M)
           CY(M)=    OCYY*Y(M)
 88    CONTINUE


 7880   FORMAT(1X,D15.8,2X,D15.8)
      CLOSE (UNIT=40)
      CALL POTEN
      RETURN
      END
C=======================================================================
C
C
C***********************************************************************
C STOCKE :                                                             *
C  Initialiser les coordonnées des atomes de bord et de l'atome active du centre .    *
C***********************************************************************
C
C===================== STOCKE ==========================================
C
C
      SUBROUTINE STOCKE
      IMPLICIT REAL*8(A-H,O-Z)
C
      COMMON/ZUT/A1,SOM,SOU,TSOM,TSOU,GAMMA,LUT,LUT1,LUT2,LA,TEST,TTEST ,LB
        COMMON/COOR/X(2601),Y(2601),Z(2601),CX(2601),CY(2601),PPX(2601),PP
       1Y(2601),VX(2601),VY(2601),VZ(2601),AX(2601),AY(2601),AZ(2601),NMOL,NMOL2,NMOL3
       COMMON/VAC/XINT(2601),YINT(2601),CXINT(2601),CYINT(2601),XVAC,
     1YVAC,CXVAC,CYVAC,DEPLCE,MVAC,LVAC
      COMMON/CELL/A1B2,A1B,A2B2,A2B,COXY,COYY,OCXY,OCYY,ALPHA
C
C Stocker X et Y dans XINT et YINT
C Remettre à zéro les  vitesses du Bord.
      DO 20  M=1,NMOL
       IF(M.LE.LA) GOTO 21
       IF(MOD(M,LA).EQ.1)  THEN
        GOTO 21
        ELSE

        GOTO 20
        ENDIF
 21     XINT(M)=X(M)
       YINT(M)=Y(M)
       CXINT(M)=CX(M)
       CYINT(M)=CY(M)
       VX(M)=0
       VY(M)=0
 20     CONTINUE
        LVAC=NMOL/2+1+LA
```

```
C Introduction du déplacement de l'atome vacante.
        DEPLCE= (MVAC*A1)/10
CC       WRITE(3,*) 'DEPLCE=',DEPLCE
      CX(LVAC)=CX(LVAC)-DEPLCE
      CY(LVAC)=CY(LVAC)
C Stocker les coordonnées de l'atome vacante dans XVAC,YVAC .
      CXVAC=CX(LVAC)
      CYVAC=CY(LVAC)
       XVAC=CXVAC+COXY*CYVAC
       YVAC=     COYY*CYVAC
      VX(LVAC)=0
      VY(LVAC)=0


      WRITE(3,200) CXVAC,CYVAC,XVAC,YVAC
200     FORMAT(' ',4(1PE15.3))
    RETURN
    END
C=========================================================================
C
C
C************************************************************************
C RESET :                                                                *
C  Sauve- garder  les coordonnées des bords et de l'atome vacante..
C************************************************************************
C
C======================= RESET ==========================================
C
C
     SUBROUTINE RESET
     IMPLICIT REAL*8(A-H,O-Z)
C
       COMMON/COOR/X(2601),Y(2601),Z(2601),CX(2601),CY(2601),PPX(2601),PP
      1Y(2601),VX(2601),VY(2601),VZ(2601),AX(2601),AY(2601),AZ(2601),NMOL,NMOL2,NMOL3
     COMMON/ZUT/A1,SOM,SOU,TSOM,TSOU,GAMMA,LUT,LUT1,LUT2,LA,TEST,TTEST ,LB
      COMMON/VAC/XINT(2601),YINT(2601),CXINT(2601),CYINT(2601),XVAC,
   1YVAC,CXVAC,CYVAC,DEPLCE,MVAC,LVAC
C Sauvegarder les coordonnées des bords et de l'atome vacante.
     DO 200  M=1,NMOL
       IF(M.LE.LA)  GOTO 201
       IF(MOD(M,LA).EQ.1) THEN
        GOTO 201
       ELSE
        GOTO  200
        ENDIF
 201     X(M)=XINT(M)
      Y(M)=YINT(M)
      CX(M)=CXINT(M)
      CY(M)=CYINT(M)
      VX(M)=0
      VY(M)=0
 200   CONTINUE
      X(LVAC)=XVAC
      Y(LVAC)=YVAC
      CX(LVAC)=CXVAC
      CY(LVAC)=CYVAC
```

```
      VX(LVAC)= 0
      VY(LVAC)= 0
      RETURN
      END
C=======================================================================
C=======================================================================
```