Feedback Control in Molecular Dynamics Simulations using LAMMPS

Loi Do and Zdeněk Hurák

***Abstract*— LAMMPS, an acronym for Large-scale Atomic and Molecular Massively Parallel Simulator, is a widely used open-source tool for high-fidelity molecular dynamics (MD) simulations. In this paper, we take the initial steps towards using LAMMPS for synthesis and validation of feedback control in nanoscale manipulation. We begin by introducing the field of MD itself, discussing the specific challenges related to control synthesis, applications of nanoscale manipulation, and the intricacies of high-fidelity MD simulations. Then, we explain the main steps in modeling a molecular system in LAMMPS and provide an illustrative example. In the example, we consider a nanoscale flake of molybdenum disulfide manipulated with the tip of an atomic force microscope over an atomic surface. We designed a simple PID controller to slide the flake with the microscope tip into a desired position. To run LAMMPS simulations with closed-loop control, we utilized the official Python wrapper for LAMMPS, upon which we implemented additional functionalities. We share the code of the simulations freely with the research community through a public repository.**

***Index Terms*— LAMMPS, feedback Control, nanoscale manipulation, molecular dynamics, atomic force microscopy**

I. Introduction

Molecular Dynamics (MD) characterizes the motion of nanoscale structures with a resolution of individual atoms. In recent decades, the MD underwent rapid development, largely attributed to the invention of Scanning Probe Microscopy (SPM) in 1981. Not only did SPM open a way to observe nanoscale structures on an atomic level, but it also enabled direct manipulation with individual atoms. An important variant of SPM is the atomic force microscope (AFM), which employs a compliant cantilever with an atomically sharp tip to probe the atoms of the material. For more detailed general description of SPM, including a control systems perspective on related technologies, we refer the reader to [1].

The advances in SPM have led to the establishment of several research fields such as *nanotribology* (study of friction and wear at nanoscale) or *nanofabrication* (manufacture of nanostructures). Naturally, these research fields have introduced many problems that could be addressed by automatic control.

In this paper, we make the first steps towards using high-fidelity MD simulations for the synthesis and validation of designed feedback control for nanoscale manipulation. As the field of MD is not widely established in the control engineering community, we also introduce the MD's challenges, motivations, and basic principles. The main contribution of this work is presenting a starting point for further research in the feedback control of MD.

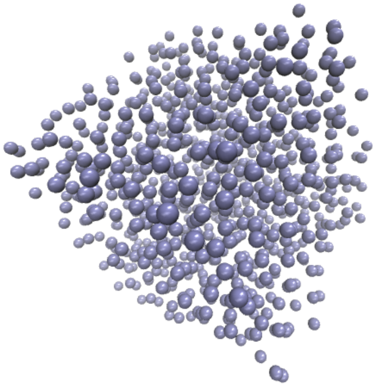

Fig. 1: Molecular dynamics simulation of argon gas contained in a box at a temperature of $94\,\mathrm{K}$

A. Challenges in Molecular Dynamics

From the perspective of control synthesis, nanoscale manipulation presents many challenges. For example, the molecular systems are generally large (thousands of atoms) with only a few degrees of freedom (DOF) that can be controlled directly, i.e., the dynamics are *underactuated* [2]. Also, the inter-atomic interactions are generally *non-linear*, and together with the high dimensionality of the system, the behavior is generally complex. Lastly, with current instrumentation, measuring the dynamical state of each atom in large MD systems is still unfeasible. Thus, the control must rely only on reduced information, i.e., aggregated measurements, such as the position and velocity of the system's center of mass or the system's temperature.

B. Motivation and Related Work

One of the promising applications of nanoscale manipulation is nanofabrication. Nanofabrication could allow further minimization of electrical devices, but also creating nanorobots, or *meta-materials*—materials with engineered properties. An essential task in nanofabrication is the manipulation of the building blocks. In [3], the authors show how a single molecule consisting of ≈ 30 atoms (representing the building block) can be extracted from a compact layer using a tip of the SPM. The translation of the building block

This work was supported by the Grant Agency of the Czech Technical University in Prague, grant No. SGS22/166/OHK3/3T/13 and by the Czech Science Foundation (GACR) under contract No. 21-07321S, and co-funded by the European Union under the project ROBOPROX (reg. No. CZ.02.01.01/00/22_008/0004590).

Loi Do and Zdeněk Hurák are with Faculty of Electrical Engineering, Czech Technical University in Prague, Technická 2, Prague 166 26, Czech Republic {doloi,hurak}[at]fel.cvut.cz

was then shown in [4]. To solve the task, the authors in both works employed reinforcement learning.

Another application of nanoscale manipulation is in a study of *nanotribology*. Using the AFM, one can slide a nanostructure over a surface while measuring and analyzing friction force. In [5], the authors made some steps towards understanding friction and energy dissipation by examining a sliding motion between two nanoscale flakes of molybdenum disulfide. An almost frictionless continual sliding between two nanomaterials—the state of *superlubricity*—was experimentally observed and analyzed in several works. For instance, superlubricity was reported for a graphene nanoribbon and gold surface [6], or for the contact of 2D heterostructures (MoS_2 and Graphite) [7]. A better understanding of friction can be exploited in the design of low-friction nanocoatings or lubricants.

Some works deal with feedback control of MD. For instance, in [8], the authors deal with atomic scale friction control by vibration. In our previous works [9], [10], we proposed that atomic scale sliding friction could be mitigated by controlling the system into *synchronization*. However, the control synthesis and analysis in the aforementioned works were based only on a simplified model of MD, the *Frenkel-Kontorova* model. In contrast, in this paper, we deal with control in MD high-fidelity simulations. For the simulations, we used the tools which are developed by and directly used within the MD community.

C. Simulation of Molecular Dynamics

There are two principal approaches in simulations of MD: a classical (Newtonian) approach and a quantum mechanical (*ab initio*) approach. In both approaches, a simulation is described by an atomic structure (geometry) of the system, i.e., the coordinates and chemical element of the atoms. The main difference is that in the classical approach, the atoms are regarded only as point masses. Thus, explicitly defining and parametrizing force interactions between the atoms is necessary. In contrast, the force interactions in the quantum-based simulations are a consequence of interactions between subatomic particles, hence directly given by the system's structure. This makes the quantum-based simulations superior to the Newtonian simulations as the quantum laws intrinsically set the system's behavior.

However, the main drawback of quantum-based simulations is significantly higher computational complexity than the Newtonian simulations. Thus, Newtonian simulations are still the prevailing approach for larger systems. In what follows, we focus only on Newtonian simulations, for which we use LAMMPS[1], an acronym for *Large-scale Atomic/Molecular Massively Parallel Simulator*. LAMMPS, distributed as an open source code, is one of the most used tools for modeling and simulation for MD, allowing simulations of systems comprising thousands of atoms; see [11] for LAMMPS overview.

II. Molecular Dynamics in LAMMPS

LAMMPS is both a modeling and simulation tool specialized for MD. A molecular system in LAMMPS is defined by an *input* script—a user-defined series of commands that characterize the model of the system, the algorithm for numerical integration, or data logging options. The MD simulation is then run by executing the input script. For further information, a list of all commands, and examples, we refer the reader to the official LAMMPS documentation[2] or many LAMMPS tutorials available online.

The main benefit of the LAMMPS is that it provides high-fidelity and fast MD simulations only by selecting suitable commands without implementing complex physical laws. However, it is still necessary to understand several concepts from MD to correctly set up a simulation. We explain these concepts in this section, together with the main steps in the simulation setup. A more detailed description of MD simulations can be found in [12].

A. Description of Dynamics

The dynamics of i-th atom in the system is given by Newton's equation of motion

$$m_i \ddot{r}_i(t) = F_i(t) \ , \tag{1}$$

where $r_i \in \mathbb{R}^3$ is the atom's coordinate, m_i is its mass, and F_i is the total force acting on the atom. The mass m_i is derived from the chemical element, specifically its atomic mass number. The total force F_i can be written as

$$F_i(t) = F_i^{\mathrm{inter}} + F_i^{\mathrm{therm}} + F_i^{\mathrm{diss}} + F_i^{\mathrm{ext}} \ , \tag{2}$$

where the individual terms account for interaction with other atoms, thermal effects, dissipation effects, and external forces, respectively. For the user-defined forces (2), the equations (1) are then numerically integrated with the integration step Δt to obtain the trajectories $r_i(t)$.

1) Interaction Force: Interactions between atoms are described by functions referred to as *potentials*. In the molecular system, the potential is defined for each pair of chemical elements. An example is the *Lennard–Jones* potential

$$E_{\mathrm{LJ}}(r_{ij}) = 4\epsilon \left[\left(\frac{\sigma}{r_{ij}} \right)^{12} - \left(\frac{\sigma}{r_{ij}} \right)^{6} \right] \ , \quad r_{ij} < r_{\mathrm{c}} \ , \tag{3}$$

where ϵ and σ are parameters, $r_{ij} = \|r_i - r_j\|_2$ is the relative distance between i-th and j-th atom, and r_{c} is cutoff distance. Let E_{tot} be the sum of all potentials in the system. The interaction force F_i^{inter} acting on i-th atom is then

$$F_i^{\mathrm{inter}} = -\nabla_{r_i} E_{\mathrm{tot}} \ , \tag{4}$$

where $\nabla_{r_i}(\cdot)$ is the gradient w.r.t. the position of i-th atom. While the Lennard–Jones potential depends on the relative position of two atoms, other potentials can account for the relative positions of multiple atoms to describe more complex behaviors.

[1] https://www.lammps.org/

[2] https://docs.lammps.org/

2) Thermal Effects: The system's temperature is directly connected to the velocity of the atoms by the *equipartition theorem*. Specifically, the relation between the system's temperature T and kinetic energy E_{kin} (hence the velocity) is

$$T = \frac{2E_{\mathrm{kin}}}{N_{\mathrm{DOF}} k_{\mathrm{B}}} \,, \tag{5}$$

where N_{DOF} is the total number of DOF, and k_{B} is the Boltzmann constant. The distribution of the total kinetic energy among individual atoms is then given by the *Maxwell–Boltzmann* distribution. When the system's kinetic energy is changed by, e.g., external forces, the system's temperature is changed, and *vice versa*.

The temperature of the system can be directly set by the *thermostats*. Thermostats model a situation when the system is in thermal contact with a large thermal bath (i.e., another system with large heat capacity and defined temperature). Internally, a thermostat is represented by a *virtual* particle (a thermal mass) interacting with all atoms. LAMMPS then automatically sets the forces F_i^{therm}, adjusting the atoms' velocities to satisfy both the equipartition theorem and the Maxwell–Boltzmann distribution.

3) Dissipation Effects: In MD simulations, dissipation forces usually account for interaction and energy exchange between the atoms and unmodeled particles. For instance, such particles can represent an implicit solvent, i.e., fluid particles, in which the system is submerged. An example of a simple dissipation model can be *viscous* damping

$$F_i^{\mathrm{diss}} = -3\pi\eta d v_i \,, \tag{6}$$

where $v_i \in \mathbb{R}^3$ is the atom's velocity, and the damping coefficient is derived from the dynamic viscosity η of the fluid and the diameter d of its particles. Different dissipation force models could also incorporate dependency on the system's temperature.

B. Simulation Setup

The starting point of a simulation is a definition of the system geometry, consisting of the size of the simulation domain, initial coordinates of the atoms $r_i(t_0)$, and their mass m_i. Since many molecules have an ordered crystal structure, they can be efficiently constructed by replication of a *unit* (*primitive*) cell. Definitions of unit cells for many molecules can be obtained from the *Materials Project* [13]. Fig. 2 shows unit cells of two molecules, MoS_2 and Graphite, together with replicated systems $(3\times3\times2)$ in x-y-z direction, respectively.

The next step is selecting the interatomic potentials for the atoms in the system. An appropriate choice of the potentials and their parameters to model the interaction between the selected chemical elements is essential for obtaining physically correct behavior. The potential selection and parametrization is usually done in a way to match the simulated behavior with experimental data or the behavior obtained from quantum-based simulations.

In contrast to simulations of macroscale dynamics, there are two additional steps in MD simulations before running the simulation: *geometry* relaxation and *thermal* equilibration. Both steps can be directly done in LAMMPS. The objective of the former step is to minimize the system's potential energy E_{tot} by adjusting the positions of the atoms. The latter aims at setting and correctly distributing the atoms' velocities—setting the desired temperature of the system in accordance with the equipartition theorem and Maxwell–Boltzmann distribution as described in Sec. II-A.2.

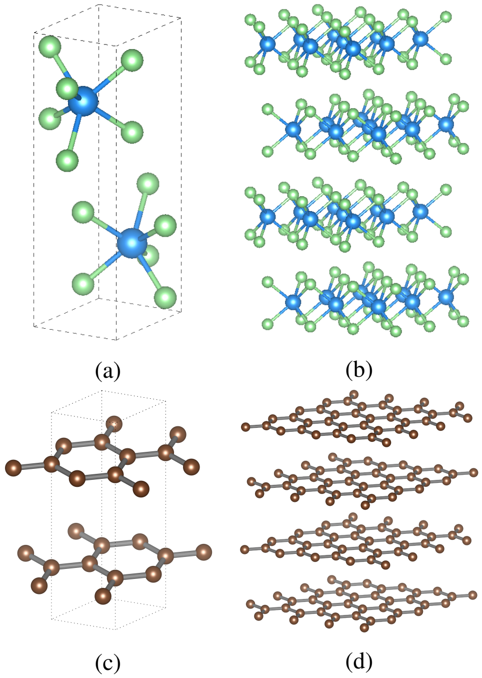

Fig. 2: Examples of unit cells: (a) MoS_2 and (c) Graphite together with $(3\times3\times2)$ replicated systems. Visualization was created with the *VESTA* software (https://jp-minerals.org/)

C. Closed-loop Control in LAMMPS

By default, LAMMPS provides commands for controlling the system only in an open loop, e.g., a command that sets the external forces F_i^{ext} to constant values throughout the whole simulation. To run LAMMPS simulations with closed-loop control, we use an official Python wrapper[3] for LAMMPS, upon which we implement additional functionalities.

When LAMMPS is run within a Python script, the simulation is represented by an *object* that encapsulates all simulation data. Python LAMMPS library then provides functions for executing the commands and reading the simulation output. The main advantage of using the Python wrapper is that the simulation can be *warm* started, i.e., setting up the simulation only once at the beginning and then periodically running the simulation for selected number of integration steps. Therefore, at every (control) period $\Delta t_{\mathrm{c}} = N\Delta t$, $N \geq 1$, we change the forces F_i^{ext} based on the simulation's output, hence effectively running the simulation with a closed-loop control. The algorithms required for closed-loop control can be easily implemented in Python.

[3] The wrapper can be found in the official LAMMPS Github repository.

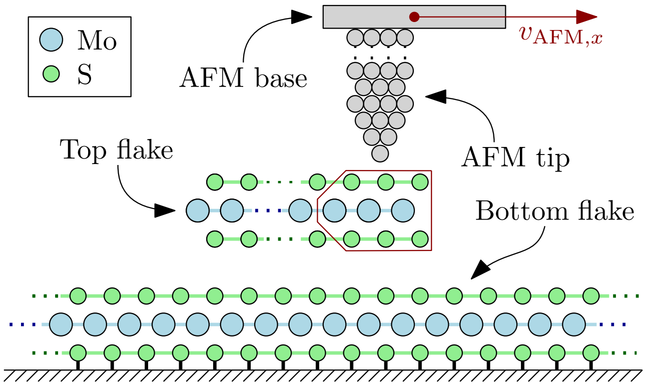

Fig. 3: Two flakes of $\mathrm{MoS_2}$ with the bottom sulfur layer fixed. Highlighted in red is the boundary part of the top flake pinned by the AFM tip (the tip is not-to-scale)

III. Example

In this section, we show an example that illustrates the basic use of LAMMPS for MD simulation and the design of a closed-loop controller to achieve a simple goal. The implementation of the simulations is available at https://github.com/aa4cc/Control-LAMMPS, including a detailed description the systems' construction and simulation parameters.

A. Sliding of $\mathrm{MoS_2}$ Flake over the $\mathrm{MoS_2}$ Substrate

We consider a system depicted in Fig. 3, adopted from [5]. The system consists of two weakly coupled *flakes* of $\mathrm{MoS_2}$ (molybdenum disulfide) with the large bottom flake being fixed, while the top flake can move freely with its edge pinned with the AFM tip. Let x_{CM} be the x coordinate of the center of mass (CM) of the top flake, $x_{\mathrm{CM}}^{\mathrm{ref}}(t)$ be a reference trajectory, and $v_{\mathrm{AFM},x}$ be a velocity of the AFM base along x axis. The task is to design a controller for $v_{\mathrm{AFM},x}$ in order to satisfy

$$\lim_{t\to\infty} |x_{\mathrm{CM}}(t) - x_{\mathrm{CM}}^{\mathrm{ref}}(t)| = 0 \, . \tag{7}$$

That is, the task is to move the top flake into a desired position by setting the velocity of the AFM base $v_{\mathrm{AFM},x}$.

1) System Initialization: To set up the simulation, we first created the system by replicating the $\mathrm{MoS_2}$ unit cell along x-y-z by $(40 \times 15 \times 1)$ and then removing $60\,\%$ of the top flake. In total, the resulting system has 2520 atoms. The top flake consists of 720 atoms, with approximate size $(4.7 \times 3.8 \times 0.3)\,\mathrm{nm}$, see Fig. 4. The bottom flake is fixed by imposing zero velocity on its bottom sulfur layer throughout the simulation.

The force interactions between molybdenum and sulfur were parametrized by the *Stillinger-Weber* potential and the interactions between sulfur atoms by *Lennard-Jones* potential, as reported in [14]. We then run the geometry relaxation, followed by the thermal equilibration of the system at $300\,\mathrm{K}$ using (a chain of three) *Nose-Hoover* thermostats for $50\,\mathrm{ps}$. The histogram of speeds (magnitudes of velocities $\|v\|_2$) of the atoms after thermal equilibration is displayed in Fig. 5. We note that although we keep the bottom sulfur layer fixed during the simulation, atoms are still thermally equilibrated with other atoms at this stage (i.e., get velocity assigned). One can check that the distributions of speeds align with the Maxwell-Boltzmann distribution.

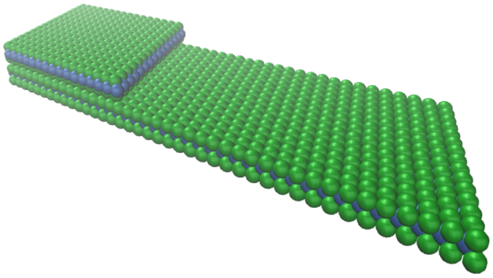

Fig. 4: Initial geometry of the system: two flakes of $\mathrm{MoS_2}$ visualized in *Visual Molecular Dynamics* software (https://www.ks.uiuc.edu/Research/vmd/)

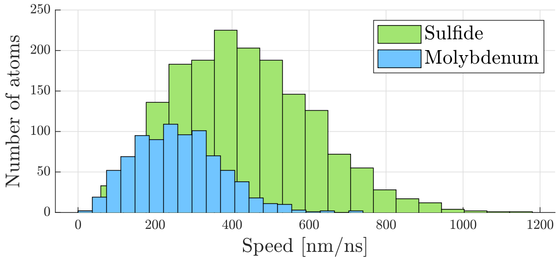

Fig. 5: Distribution of atoms' speed $\|v\|_2$ at $T = 300\,\mathrm{K}$, following the Maxwell–Boltzmann distribution

2) Thermal and Dissipation effects: Throughout the simulation, we model the thermal and dissipation effects using the *Langevin* thermostat that sets the forces in (2) as

$$F_i^{\mathrm{diss}} = -\frac{m_i}{\gamma} v_i \ , \quad F_i^{\mathrm{therm}} \propto \sqrt{\frac{k_{\mathrm{B}} T m_i}{\gamma \Delta t}} \ , \tag{8}$$

where γ is the damping parameter that determines the speed of temperature relaxation. The term F_i^{diss} represents viscous damping, while the second term F_i^{therm} represents implicit solvent at temperature T randomly bumping into the atoms. The exact value and direction of F_i^{therm} is randomized.

3) Actuation through AFM Tip: Similarly to [6], [15], we represent the AFM tip by a linear spring with one end attached to a virtual CM of the pinned atoms, and with the second end moving with the velocity $v_{\mathrm{AFM},x}(t)$, see Fig 6. In particular, let $\mathcal{S}$ be the set of indices of the atoms in contact with the AFM tip, i.e., directly pulled by the spring. Furthermore, let $x_{\mathcal{S}}(t)$ be the position of CM of the atoms belonging to $\mathcal{S}$. Then, the external force acting on each pulled atom is

$$F_i^{\mathrm{ext}} = k \left(x_{\mathrm{AFM}}(t) - x_{\mathcal{S}}(t) \right) \frac{m_i}{\sum_{j \in \mathcal{S}} m_j} \ , \quad i \in \mathcal{S} \ , \tag{9}$$

where k is the spring's stiffness and

$$x_{\mathrm{AFM}}(t) = x_{\mathrm{AFM}}(t_0) + \int_{t_0}^{t} v_{\mathrm{AFM},x}(\tau) \mathrm{d}\tau \ , \tag{10}$$

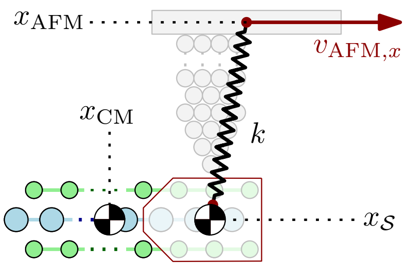

Fig. 6: AFM tip represented by a linear spring with a stiffness k pulling the atoms that belong to the set $\mathcal{S}$

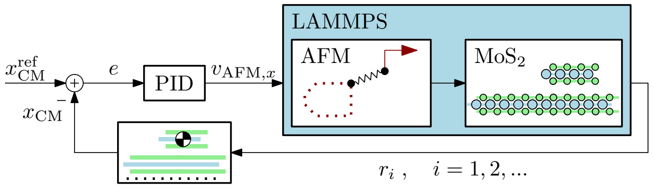

Fig. 7: Block diagram of the control loop. The goal is to drive the top's flake center of mass x_{CM} to the reference position $x_{\mathrm{CM}}^{\mathrm{ref}}$

with $x_{\mathrm{AFM}}(t_0)$ being the AFM tip's initial position at t_0.

4) Control Design: We set the output of the system to be only the position of the top flake's CM $x_{\mathrm{CM}}(t)$ and the input to the system to be $v_{\mathrm{AFM},x}$. To deal with the defined problem (7), we design a PID controller that sets the velocity of the AFM tip

$$v_{\mathrm{AFM},x}(t) = k_{\mathrm{P}} e(t) + k_{\mathrm{I}} \int_{t_0}^{t} e(\tau)\mathrm{d}\tau + k_{\mathrm{D}} \frac{\mathrm{d}}{\mathrm{d}t} e(t)\ , \tag{11}$$

where $e(t) = x_{\mathrm{CM}}^{\mathrm{ref}}(t) - x_{\mathrm{CM}}(t)$, and $k_{\mathrm{P}}, k_{\mathrm{I}}, k_{\mathrm{D}}$ are PID design constants. The velocity of the AFM tip $v_{\mathrm{AFM},x}$ is changed periodically with Δt_{c}. The block diagram of the control loop is in Fig. 7. For implementation, the control (11) is then discretized with a sampling time Δt_{c}.

We tuned the PID constants by running multiple simulations to achieve the desired performance. In particular, the constants were set so the velocity $v_{\mathrm{AFM},x}$ does not exceed[4] $\pm 1\,\mathrm{m\,s^{-1}}$, since too high pulling velocity could result in break of the flake, see Fig. 8 for an illustration. Another consideration in tuning the PID constants was to minimize the overshoot of $x_{\mathrm{CM}}(t)$ w.r.t. to reference $x_{\mathrm{CM}}^{\mathrm{ref}}(t)$.

5) Simulation: All relevant simulation parameters are listed in Tab. I. We attached the spring representing the AFM tip to the rightmost $(3 \times 15 \times 3)$ block of atoms in the top flake. The total simulation time was $20\,\mathrm{ns}$ which was executed in approx. $90\,\mathrm{min}$ on a standard desktop computer with eight CPU cores. The resulting simulation is in Fig. 9 with an image of the end position in Fig. 10. We can see, that $x_{\mathrm{CM}}(t)$ is successfully driven to the desired position $x_{\mathrm{CM}}^{\mathrm{ref}}(t)$.

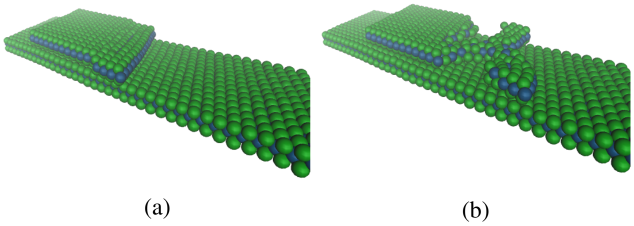

Fig. 8: Snapshots of two consecutive moments from the simulation displaying breaking of the top flake due to high pulling velocity

TABLE I: Simulation parameters

Description	Symbol	Value
Integration step	Δt	$15\,\mathrm{fs}$
Control step	Δt_{c}	$50\,\mathrm{ps}$
Spring's stiffness	k	$80\,\mathrm{N\,m^{-1}}$
Spring's init. elong.	$x_{\mathrm{AFM}}(t_0)$	$0.6\,\mathrm{nm}$
System's temperature	T	$300\,\mathrm{K}$
Damping parameter	γ	$50\,\mathrm{fs^{-1}}$
PID constants	k_{P}	1.9×10^{-7}
	k_{I}	1.5×10^{-4}
	k_{D}	1×10^{-2}

An interesting phenomenon, which can be observed in Fig. 9, is the *stick-slip* motion of the flake. Due to force interactions between the flakes, the top flake *sticks* in a particular position until the force exerted by the spring exceeds the inter-flake force, resulting in *slip*. The *stick-slip* motion depends on many variables, such as system's temperature or commensurability of the layers in contact, and might be one of the key causes of friction, see [16] for more details.

IV. Conclusion and Outline

In this paper, we presented the first steps towards using LAMMPS for synthesis and validation of feedback control for nanoscale manipulation. We extended the official Python wrapper for LAMMPS to allow closed-loop control of the simulation. To demonstrate the developed framework, we showed an example with a high-fidelity simulation of a MoS$_2$ flake pinned with the tip of the atomic force microscope. We designed a basic PID controller that slides the flake into a desired position. The code of the simulations are freely shared with the research community through a public repository.

The future work will focus on more advanced control. In particular, we proposed in our previous works [9], [10] that by controlling the system into achieving synchronization in atoms' states, we can reduce sliding friction by mitigating the stick-slip motion. For control synthesis, we will use the *Koopman Model Predictive Control* [17] that combines data-driven system identification with optimization-based control.

Acknowledgements

We would like to thank Antonio Cammarata for his valuable comments on this work.

[4] In real experiments, the microscope's operational velocity is usually in the order of nanometers per second. Although we use significantly higher velocity in our simulation, the simulation still preserves important dynamical features, while the simulation is less computationally expensive.

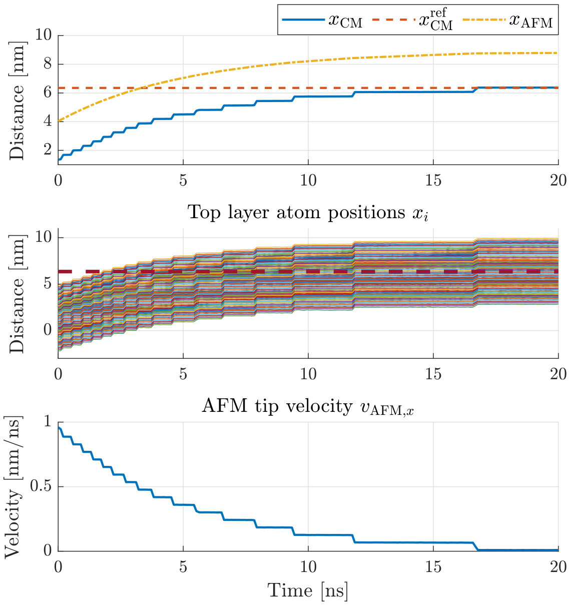

Fig. 9: Simulation result of moving the top MoS_2 flake into the desired position with PID controller. The figure displays (from the top to the bottom): aggregated quantities; x-coordinates of all atoms in the top layer; the control input

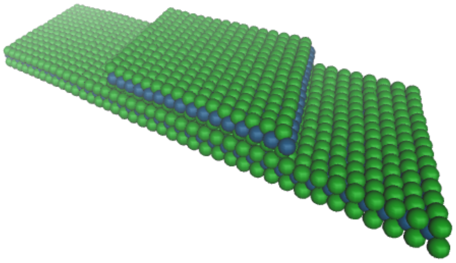

Fig. 10: Visualization of the system at the end of the simulation. The top MoS_2 flake was successfully moved into the desired position

References

[1] S. M. Salapaka and M. V. Salapaka, "Scanning Probe Microscopy," *IEEE Control Systems Magazine*, vol. 28, no. 2, pp. 65–83, Apr. 2008.

[2] R. Tedrake, *Underactuated robotics*, 2009. [Online]. Available: https://underactuated.csail.mit.edu

[3] P. Leinen, M. Esders, K. T. Schütt, C. Wagner, K.-R. Müller, and F. S. Tautz, "Autonomous robotic nanofabrication with reinforcement learning," *Science Advances*, vol. 6, no. 36, pp. 1–8, Sep. 2020.

[4] B. Ramsauer, G. J. Simpson, J. J. Cartus, A. Jeindl, V. García-López, J. M. Tour, L. Grill, and O. T. Hofmann, "Autonomous Single-Molecule Manipulation Based on Reinforcement Learning," *The Journal of Physical Chemistry A*, vol. 127, no. 8, pp. 2041–2050, Mar. 2023, publisher: American Chemical Society.

[5] A. Cammarata, P. Nicolini, K. Simonovic, E. Ukraintsev, and T. Polcar, "Atomic-scale design of friction and energy dissipation," *Physical Review B*, vol. 99, no. 9, p. 094309, Mar. 2019, publisher: American Physical Society.

[6] S. Kawai, A. Benassi, E. Gnecco, H. Söde, R. Pawlak, X. Feng, K. Müllen, D. Passerone, C. A. Pignedoli, P. Ruffieux, R. Fasel, and E. Meyer, "Superlubricity of graphene nanoribbons on gold surfaces," *Science*, vol. 351, no. 6276, pp. 957–961, Feb. 2016, publisher: American Association for the Advancement of Science.

[7] M. Liao, P. Nicolini, L. Du, J. Yuan, S. Wang, H. Yu, J. Tang, P. Cheng, K. Watanabe, T. Taniguchi, L. Gu, V. E. P. Claerbout, A. Silva, D. Kramer, T. Polcar, R. Yang, D. Shi, and G. Zhang, "Ultra-low friction and edge-pinning effect in large-lattice-mismatch van der Waals heterostructures," *Nature Materials*, vol. 21, no. 1, pp. 47–53, Jan. 2022, number: 1 Publisher: Nature Publishing Group.

[8] Y. Guo, Z. Wang, Z. Qu, and Y. Braiman, "Atomic-scale friction control by vibration using friction force microscope," *Control Engineering Practice*, vol. 19, no. 11, pp. 1387–1397, Nov. 2011.

[9] L. Do and Z. Hurák, "Synchronization in the Frenkel-Kontorova Model with Application to Control of Nanoscale Friction," *IFAC-PapersOnLine*, vol. 54, no. 14, pp. 406–411, Jan. 2021.

[10] L. Do, M. Korda, and Z. Hurák, "Controlled synchronization of coupled pendulums by Koopman Model Predictive Control," *Control Engineering Practice*, vol. 139, p. 105629, Oct. 2023.

[11] A. P. Thompson, H. M. Aktulga, R. Berger, D. S. Bolintineanu, W. M. Brown, P. S. Crozier, P. J. in 't Veld, A. Kohlmeyer, S. G. Moore, T. D. Nguyen, R. Shan, M. J. Stevens, J. Tranchida, C. Trott, and S. J. Plimpton, "LAMMPS - a flexible simulation tool for particle-based materials modeling at the atomic, meso, and continuum scales," *Computer Physics Communications*, vol. 271, p. 108171, Feb. 2022.

[12] D. Frenkel and B. Smit, *Understanding Molecular Simulation: From Algorithms to Applications*, 2nd ed. San Diego: Academic Press, Nov. 2001.

[13] A. Jain, S. P. Ong, G. Hautier, W. Chen, W. D. Richards, S. Dacek, S. Cholia, D. Gunter, D. Skinner, G. Ceder, and K. A. Persson, "Commentary: The Materials Project: A materials genome approach to accelerating materials innovation," *APL Materials*, vol. 1, no. 1, p. 011002, Jul. 2013.

[14] J.-W. Jiang, H. S. Park, and T. Rabczuk, "Molecular dynamics simulations of single-layer molybdenum disulphide (MoS2): Stillinger-Weber parametrization, mechanical properties, and thermal conductivity," *Journal of Applied Physics*, vol. 114, no. 6, p. 064307, Aug. 2013.

[15] W. Ouyang, D. Mandelli, M. Urbakh, and O. Hod, "Nanoserpents: Graphene Nanoribbon Motion on Two-Dimensional Hexagonal Materials," *Nano Letters*, vol. 18, no. 9, pp. 6009–6016, Sep. 2018, publisher: American Chemical Society.

[16] S. Y. Krylov and J. W. M. Frenken, "The physics of atomic-scale friction: Basic considerations and open questions," *physica status solidi (b)*, vol. 251, no. 4, pp. 711–736, 2014.

[17] M. Korda and I. Mezić, "Linear predictors for nonlinear dynamical systems: Koopman operator meets model predictive control," *Automatica*, vol. 93, pp. 149–160, Jul. 2018.