ISSN 1392-1207. MECHANIKA. Vol. 32, No. 5, 2026: 420–426

Numerical Implementation of a Jamming Gripper via Discrete Element Method in LAMMPS

Shintaro HOKADA*, Satoshi TAKADA**
**Department of Industrial Technology and Innovation, Tokyo University of Agriculture and Technology, 2-24-16, Nakacho, Koganei, Tokyo, 184-8588, Japan, E-mail: hokada@st.go.tuat.ac.jp*
***Department of Mechanical Systems Engineering, Tokyo University of Agriculture and Technology, 2-24-16, Nakacho, Koganei, Tokyo, 184-8588, Japan, E-mail: takada@go.tuat.ac.jp*
https://doi.org/10.5755/j02.mech.44049

1. Introduction

Techniques for grasping objects accurately and stably play a fundamental and essential role in a wide range of fields, including industrial robotics, medical devices, and disaster-response systems. Conventional robotic hands typically employ multiple finger-like structures analogous to the human hand, incorporating various sensors and actuators and relying on sophisticated control schemes to achieve dexterous manipulation. However, such hands require task-specific design and control depending on the shape and stiffness of the target object, and therefore often function effectively only in limited environments or for specific types of objects.

Against this background, recent years have seen growing interest in the development of grasping mechanisms that offer flexibility and adaptability. In particular, soft robotic hands have been widely studied for handling objects with complex geometries or compliant surfaces [1]. Examples of soft grippers include jam-ming grippers [2-5], suction-based grippers [6, 7], and finger-type soft grippers [8-10]. In the present study, we focus on jamming grippers.

Granular materials, composed of discrete particles, exhibit a phenomenon known as the jamming transition, in which their macroscopic behavior switches between solid-like and fluid-like states depending on changes in packing density [11]. Jamming grippers exploit this property. As illustrated in Fig. 1 and described in Ref. [2], a jamming gripper consists of a membrane filled with granular particles. When the membrane is pressed onto a target object, it deforms to conform to the object's shape; subsequently, the granular medium is densified, typically by applying negative pressure, and undergoes a jamming transition, resulting in a rigid configuration that securely grasps the object.

This mechanism enables a simple internal structure consisting only of a compliant membrane and granular particles, while the control strategy is limited to operations such as adjusting internal pressure or applying magnetic fields to manipulate particle density. Despite this simplicity, jamming grippers are capable of grasping a wide variety of objects.

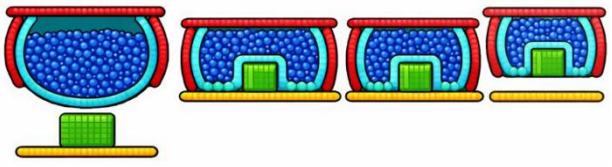


Fig. 1 Schematic of the grasping behavior of a jamming gripper: (from left to right) the gripper approaches the object, deforms to envelop the object, firmly grasps it by reducing the internal pressure, and finally lifts the object (based on [2])

Previous studies have investigated the relationship between various design elements of jamming grippers and their grasping performance through experiments and simulations [12-16], and several novel gripper designs have been proposed [3, 17]. Although these works have achieved meaningful progress, a complete understanding of the grasping conditions has not yet been established. This difficulty stems partly from the intrinsic characteristics of jamming grippers. In contrast to conventional robotic hands, whose grasping behavior can be explicitly controlled by adjusting finger positions, actuation forces, or material properties, jamming grippers possess only two primary control parameters: the indentation depth applied to the object and the internal pressure of the membrane. All other aspects of the grasping behavior occur passively through physical interactions among the particles, membrane, and target object. As a consequence, the performance of jamming grippers is highly sensitive to design parameters.

Moreover, because jamming grippers can deform to conform to a wide range of object geometries, establishing generalized grasping conditions requires exploring a broad set of design parameters and object shapes. In experimental settings, this would necessitate manufacturing numerous gripper prototypes and test objects, resulting in significant cost and effort. Although some simulation-based studies exist, they often focus on contact models, geometric formulations, or theoretical aspects, while concrete implementation procedures or executable code for reproducing the jamming behavior are rarely provided.

To address this issue, the present paper proposes a practical DEM-based simulation framework for reproducing the behavior of jamming grippers. The main objective is not to develop a new DEM solver, but to establish an easily reproducible numerical procedure for implementing the essential components of a jamming gripper, including the granular medium, deformable membrane, depressurization process, and rigid objects. Such a framework can reduce the effort required for repeated prototyping and provide a basis for systematically investigating the effects of particle properties, gripper geometry, and operating conditions. Importantly, the entire implementation is realized using standard functions available in existing DEM software, without modifying its source code or developing additional packages. In the present study, LAMMPS [18] is employed as an example implementation, and the input scripts used for the

simulations are provided. This approach makes the proposed framework readily reproducible and potentially transferable to other DEM platforms with comparable functionalities.

2. Model

As described in the previous section, the jamming transition is reproduced in this study by numerically integrating the equations of motion for all particles using the discrete element method (DEM) [19, 20]. For the present implementation, we employ the Large-scale Atomic/Molecular Massively Parallel Simulator (LAMMPS) [18, 21], an open-source molecular and granular dynamics simulator. LAMMPS was selected because it provides, within a single framework, the particle-contact, bonded-interaction, rigid-body, and user-defined force functionalities required for the present jamming-gripper model. In addition, these operations can be combined through input-script commands without modifying the source code. This feature is particularly useful for the present purpose because our aim is to provide a reproducible implementation that can be readily modified for different gripper geometries and operating conditions. Other DEM packages, such as EDEM and YADE, can also be used to simulate granular systems. This section describes the interaction model and the implementation of the mechanisms required to reproduce the behavior of a jamming gripper within LAMMPS.

2.1. Interactions

We first summarize the interaction model used in this work. Let r_i, v_i, and θ_i denote the position, velocity, and angular displacement vectors of particle i, respectively. In addition, the mass and diameter of particle i, are, respectively, written as m_i, and d_i.

When two particles i and j are in contact, the force acting on particle i is assumed to be

$$F_{ij} = F_{ij}^n n_{ij} + F_{ij}^t t_{ij}, \tag{1}$$

where η_{ij} and t_{ij} are the unit vectors in the normal and tangential directions at the contact point. The normal and tangential force components, F_{ij}^n and F_{ij}^t follow the Hertz–Mindlin contact model [22]:

$$\begin{aligned} F_{ij}^n &= \sqrt{R_{ij}\delta_{ij}}\left(k_n\delta_{ij} - m_{ij}\gamma_n v_{ij}^n\right), \\ F_{ij}^t &= -\left[\mu F_{ij}^n,\ \sqrt{R_{ij}\delta_{ij}}\left(k_t \Delta s_{ij}^t + m_{ij}\gamma_t v_{ij}^t\right)\right]. \end{aligned} \tag{2}$$

Here, k_n and k_t are the nonlinear stiffness coefficients in the normal and tangential directions, while γ_n and γ_t are the corresponding damping coefficients and μ is the friction coefficient. The quantities appearing in the model are defined as

$$m_{ij} = \frac{m_i m_j}{m_i + m_j},$$

$$R_{ij} = \frac{d_i d_j}{2\left(d_i + d_j\right)},$$

$$\delta_{ij} = \frac{d_i + d_j}{2} - r_{ij}, \tag{3}$$

where m_{ij} is the reduced mass, R_{ij} is the effective radius, and δ_{ij} is the overlap between particles. The variable Δs_{ij}^t denotes the tangential displacement integrated from the onset of contact.

The translational and rotational dynamics of particle i evolve according to

$$m_i \frac{d^2 r_i}{dt^2} = \sum_{j\neq i} F_{ij},\ I_i \frac{d^2\theta_i}{dt^2} = \sum_{j\neq i} T_{ij}, \tag{4}$$

where $I_i = m_i d_i^2 / 10$ is the moment of inertia and

$$T_{ij} = \frac{d_i}{2} n_{ij} \times F_{ij} \tag{5}$$

is the torque exerted on particle i due to particle j.

2.2. Elastic membrane

A key component of the jamming gripper is the elastic membrane surrounding the granular particles. By reducing the pressure inside the membrane, a pressure difference is created, compressing the particle assembly and inducing a jamming transition. Although membranes are typically analyzed using the finite element method, implementing such an elastic structure in a DEM-based simulator like LAMMPS presents two challenges: 1. reproducing the membrane itself, and 2. reproducing the depressurization process. This subsection focuses on the construction of the membrane.

Since DEM inherently represents particles as rigid bodies, it does not directly model deformable continua such as elastic membranes. To approximate membrane elasticity, we utilize pairwise bonding interactions. LAMMPS provides several bonded interaction models in addition to the Hertzian contact force; among them, we use the bpm/spring bond, which behaves like a spring connecting two particles. Unlike the Hertz contact model, this bonded interaction produces attractive forces when bonded particles separate and repulsive forces when they move too close. By arranging particles in a sheet-like structure and bonding neighboring particles, we can construct a membrane-like surface. Although this approach does not achieve finite-element accuracy, it is sufficient for confining the granular material within the gripper.

However, because this bonding method requires manually specifying pairs of bonded particles, generating a spherical membrane suitable for the jamming gripper is nontrivial. To automate this process, we employ Blender [23] to create a triangular-surface 3D model of a sphere and extract the vertex and edge information for particle placement and bonding. The spherical membrane is constructed by placing particles on a sphere of radius R_{mem}, and triangular elements are required for implementing the depressurization mechanism described later.

The sphere is generated via iterative subdivision of an icosahedron. Starting from a regular icosahedron, each triangular face is subdivided into four smaller triangles by bisecting each edge, and all vertices are projected onto the

circumscribed sphere. This refinement process corresponds to the creation of an “ico sphere” in Blender. We perform four subdivision steps, resulting in a mesh with 5120 faces and 2562 vertices (Fig. 2).

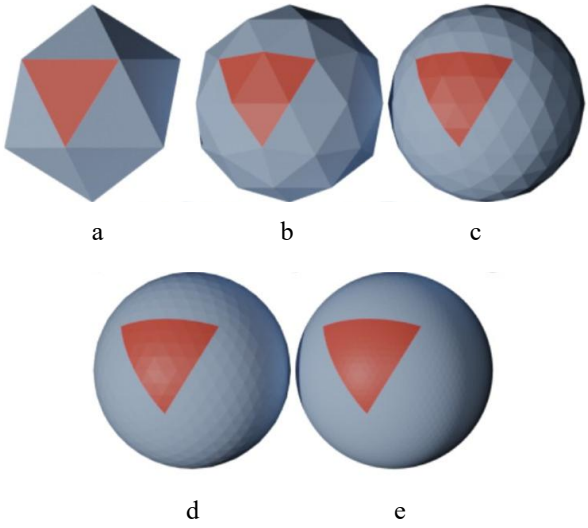


Fig. 2 Generation process of a sphere composed of triangular elements: a – the icosahedron, b – 80 faces, c – 320 faces, d – 1280 faces, and e – 5120 faces

To create an opening for particle insertion and to install a sup-porting plate that restricts upward deformation, the upper region of the sphere is removed. To ensure smoothness of the cut surface, the removed region is chosen as the five faces sharing the top vertex in the original icosahedron. Fig. 3 highlights the re-moved faces on both the sphere and the underlying icosahedron. The vertices along the boundary of the cut are treated as the membrane rim and are fixed during the simulation to support the gripper.

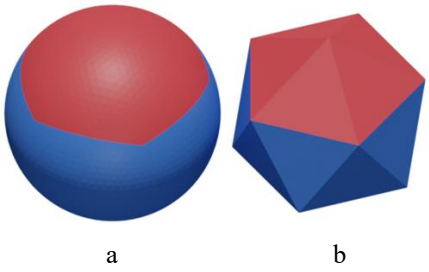


Fig. 3 Faces removed for plate placement in the spherical model (a) and the icosahedral base mesh (b)

From the 3D model, we extract the vertex coordinates and edge connectivity information and use them to generate the LAMMPS input script defining the elastic membrane. The mem-brane particles are inserted into the simulation using the **deposit** command. To facilitate the depressurization mechanism, the vertex indices of the 3D model are assigned directly as the particle IDs in LAMMPS.

For the membrane particles, only the bonded spring interactions are applied; contact forces between membrane particles are disabled. In contrast, interactions between membrane particles and other particles follow the Hertz–Mindlin model described earlier. This choice is necessary because the membrane particles must be sufficiently large to prevent granular particles from leaking through gaps in the membrane, but direct Hertzian contact between membrane particles would inhibit deformation. By retaining only bonded elastic interactions among membrane particles and using contact interactions only with other particle types, we construct an elastic yet permeation-resistant membrane suitable for the jamming gripper model.

2.3. Depressurization process

We next describe the depressurization procedure, which constitutes the second major challenge in modeling the elastic membrane. As noted earlier, the elastic membrane in this simulation is not an actual continuous membrane but a set of particles connected by special bonded interactions. Consequently, the concepts of “inside” and “outside” or a distinct membrane surface do not inherently exist, and LAMMPS does not provide a built-in function to reduce pressure inside such a particle-based structure. Modifying the LAMMPS source code to detect interior regions and apply pressure differences is impractical.

Therefore, in this study, the pressure-induced forces on membrane particles are explicitly computed within the input script. The derivation of the force calculation method is explained below.

During depressurization, pressure acts on each triangular face constituting the membrane. Since nonplanarity can occur for polygons with more than three vertices, the membrane is constructed exclusively from triangular faces. The force generated by a pressure difference acts normal to the face, with magnitude proportional to the face area. In three-dimensional space, both the area and normal vector of a triangular face can be obtained using the cross product of two edge vectors: half the magnitude of the cross product gives the triangular area, and the cross-product direction gives its normal.

For a triangular face A_{ijk} formed by particles i, j, and k, the force $F_{A_{ijk}}$ acting at its centroid due to a pressure difference ΔP is computed as

$$F_A = \frac{\Delta P}{2}\left(r_j - r_i\right)\times\left(r_k - r_i\right). \tag{6}$$

This force is then equally distributed among the three vertices of the triangle:

$$F_{A,i} = F_{A,j} = F_{A,k} = \frac{F_{A_{ijk}}}{3}. \tag{7}$$

Because the orientation of the normal vector depends on the ordering of i, j, and k, the direction may be reversed if the vertices are listed in the opposite order. Detecting correct orientations dynamically during large deformations is not feasible; thus, we determine the inward-facing vertex order in advance during preprocessing, as described later. By applying this calculation to every triangular face and summing the contributions, the total pressure-induced force acting on each membrane particle is obtained. For a particle i belonging to multiple faces $A_{ij'k'}$, the total force is

$$F_i^p = \sum_{j',k'} \frac{F_{A_{ij'k'}}}{3}. \tag{8}$$

In LAMMPS, applying particle-specific operations such as individual forces is generally done by defining a group that contains only the target particle. However, the number of groups that can be simultaneously defined is limited to 32, making this approach infeasible for the nearly 2000 membrane particles used in this work. To overcome this limitation, we introduce a pseudo-step-function based on particle IDs. A force intended for particle i, denoted by F_i, is assigned using

$$F_i\left(ID\right) = F_i a^{-(i-ID)^2} , \tag{9}$$

where *ID* denotes the particle ID and a is a sufficiently large constant (set to 10^{10} in this study). For $ID = i$, the exponential term becomes unity, and the full intended force is applied. For any other particle, the multiplier becomes at most 10^{-10}, resulting in a negligible applied force. By evaluating this expression for all particles, we effectively apply particle-specific forces without exceeding the group limit.

Implementation within LAMMPS requires the following preparations: A- extraction of vertex triplets defining each triangular face from the Blender 3D model, b- extraction of the inward-facing vertex order for each face. Using this information, each membrane particle determines the faces to which it belongs and the *ID*s of the corresponding vertices. In the input script, the following operations are performed repeatedly: A- reading particle positions, B- computing forces using the predefined vertex order, C- summing the contributions from all faces containing the particle, D- applying the resulting force to the particle, E- repeating steps A-D periodically. Details of the implementation are illustrated below.

The preprocessing (steps a- and b-) are performed in Blender. Blender stores, for each mesh face, the vertex *ID*s and its orientation. By extracting the vertex order directly, we obtain the right-hand-rule ordering that corresponds to outward normals. Because particle *ID*s are matched to vertex *ID*s, simply reordering the vertices accordingly ensures that all computed normals point inward.

Table 1

LAMMPS input script fragment for the implementation of the depressurization process

```
run 10000 every 100 &
   "variable x1 equal 0"&
   "variable y1 equal 0"&
   "variable z1 equal 0"&
   "variable x1_1 equal (y[27]-y[1])*(z[26]-z[1])-(z[27]-z[1])*(y[26]-y[1])"&

   "variable y1_1 equal (z[27]-z[1])*(x[26]-x[1])-(x[27]-x[1])*(z[26]-z[1])"&
   "variable z1_1 equal (x[27]-x[1])*(y[26]-y[1])-(y[27]-y[1])*(x[26]-x[1])"&
   "variable x1 equal ${x1}+${x1_1}"&
   "variable y1 equal ${y1}+${y1_1}"&
   "variable z1 equal ${z1}+${z1_1}"&
   ...
   "unfix force1"&
   "variable fx1 atom ${x1}*${large_num}^(-((1+${small_num}-id)^2))"&
   "variable fy1 atom ${y1}*${large_num}^(-((1+${small_num}-id)^2))"&
   "variable fz1 atom ${z1}*${large_num}^(-((1+${small_num}-id)^2))"&
   "fix force1 mem_particles addforce v_fx1 v_fy1 v_fz1"&
   ...
```

The LAMMPS input script fragment for this implementation is shown in Table 1. The command **run 10000 every 100** corresponds to step (E). In LAMMPS, **run 10000** advances the simulation by 10,000 timesteps, while the keyword **every 100** specifies commands executed at intervals of 100 timesteps. Multiple commands can be executed at each interval by enclosing each in quotation marks. Lines ending with & are treated as continuations of the same command.

In the first half of the loop, steps (A)-(C) are performed. The example shows the beginning of calculations for particle 1. Variables ***x*1**, ***y*1**, and ***z*1** are initialized and incremented by the components of the normal vector for the triangular face composed of particles 1, 26, and 27. Variables are defined using the syntax **variable name equal expression**, and particle coordinates are obtained as ***x*[*i*]**, ***y*[*i*]**, and ***z*[*i*]**.

In the latter half, step (D) is executed. In LAMMPS, fix is used to apply forces, and **unfix** removes previous assignments. After clearing the previous force, the atom-style variables ***fx*1**, ***fy*1**, and ***fz*1** are defined following Eq. (9). The **atom** keyword is used so that each particle evaluates the expression with its own *ID*. The constants **large_num** and **small_num** correspond to a and a small offset to avoid zero exponents (which cause errors in LAMMPS).

Finally, the fix **addforce** command applies the computed force as follows:

fix force1 mem_particles addforce v_fx1 v_fy1 v_fz1

where **mem_particles** is the group containing all membrane particles. Variables that evaluate to a single value are referenced as **${name}**, whereas per-atom variables are referenced as **v_name**.

2.4. Rigid bodies constructed from particles

Rigid components such as the grasped object, the gripper lid, and the cover are constructed as aggregates of particles. LAMMPS provides the **fix rigid** command to treat a group of particles as a single rigid body. For each rigid component, particles are arranged to match the desired shape and are then grouped into one rigid body.

Complex shapes such as the cover and grasped object are first modeled in Blender, and the vertex coordinates are exported. These coordinates are introduced into

LAMMPS using the **read_data** command. Unlike the **deposit** command, which inserts predefined molecules randomly or within specified regions, **read_data** places particles at exact coordinates, making it more appropriate for constructing rigid objects.

Furthermore, by setting interactions between these rigid-body particles and all other particles to zero, they effectively behave as rigid walls. In this study, the gripper lid and cover are fixed using this method.

2.5. Simulation setup

The overall simulation setup is shown in Fig. 4. The detailed parameters, particle sizes, masses, gravitational acceleration, and nondimensionalized quantities used in the simulation are summarized in Table 2.

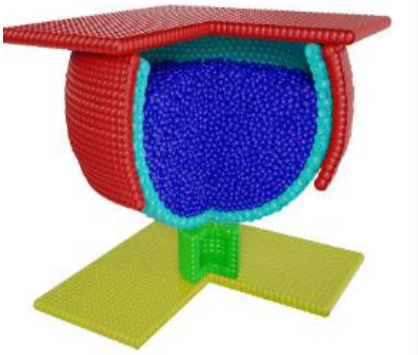


Fig. 4 Overview of the constructed simulation setup

Table 2

List of fixed simulation parameters

Symbol	Value	Unit	Dimensionless value
ρ	1.00×10^{3}	kg/m^3	1
m	4.189×10^{-3}	kg	4.189
m_{mem}	4.189×10^{-3}	kg	4.189
d	1.00×10^{-2}	m	1
d_{mem}	2.00×10^{-2}	m	2
d_{obj}	2.00×10^{-2}	m	2
R_{mem}	17.3×10^{-2}	m	17.3
l_{obj}	4.50×10^{-2}	m	4.50
g	9.81	m/s^2	1
$\mu_{p\text{-}p}$	0.1	–	0.1
μ	1.0	–	1.0
N	2000	–	2000
N_{mem}	1961	–	1961
ΔP	1.00×10^{5}	N/m^2	1.00×10^{3}
F	2.943	N	300
Δt	1.00×10^{-6}	s	1.00×10^{-4}

The spherical elastic membrane described in Sect. 2.2 encloses the granular particles of diameter d and mass m. A rigid lid and side cover, modeled according to Sect. 2.4, are placed to close the opening and prevent detachment. Finally, a cylindrical target object and a floor, both constructed from particle assemblies, are added to complete the setup. The individual components are indicated in Fig. 4.

2.6. Simulation procedure

The simulation proceeds from setup to grasping according to the following steps:

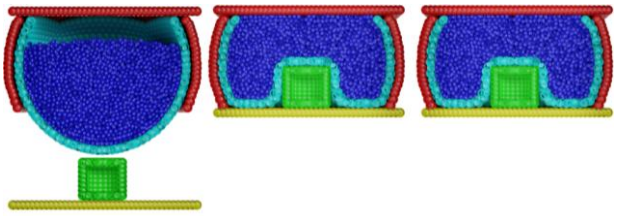
1. Form the elastic membrane by bonding N_{mem} particles of radius R_{mem} and mass m_{mem}.
2. Insert N inner particles of diameter d and mass m into the membrane from above.
3. Place the grasped object along the vertical axis passing through the membrane center, and position the floor beneath it.
4. Install the lid and side cover onto the membrane.
5. Press the object upward against the gripper with a force F_{set}.
6. Depressurize the interior of the membrane to induce the jamming transition.
7. Lower the floor and observe whether the grasped object remains held or detaches.

The floor is moved rather than the gripper to reduce the number of components requiring active control. If the gripper were moved downward, forces would need to be applied to the lid, cover, and membrane. In contrast, pressing the object upward requires only the controlled motion of the floor.

3. Results

In this section, we describe the behavior of the jamming gripper observed in the simulations conducted in this study. The operation of the jamming gripper during successful grasping is presented according to the processes outlined in Sect. 2.6. For clarity, cross-sectional views along the central depth of the gripper are shown.

Fig. 5 shows the state of the jamming gripper at the end of Processes 4, 5, and 6 explained in Sect. 2.6. This stage represents the completion of the setup, including the formation of the elastic membrane, filling of internal particles, and placement of the base, cover, and target object. As shown in Fig. 5, a, a small gap is intentionally left at the top of the gripper by controlling the number of particles, allowing the target object to be pressed in subsequent processes.



a b c

Fig. 5 Overviews of the jamming gripper at the end of Processes 4 (a), 5 (b), and 6 (c) explained in Sect. 2.6

Fig. 5, b shows the state of the jamming gripper at the end of Process 5. The base and target object have risen until the upward force applied to the base is balanced by the reaction from the gripper. The gap observed at the top of the gripper in Fig. 5, a has closed, and the elastic membrane conforms to the cover, base, and target object.

Fig. 5, c presents the state of the jamming gripper at the end of Process 6. During depressurization, the granular particles are compressed as the membrane contracts. Although the resulting change in the internal particle configuration is not easily visible in Fig. 5, c, the increased confinement restricts particle rearrangement and stabilizes the deformed configuration of the gripper.

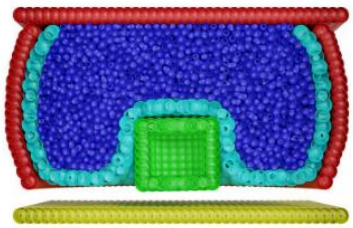


Fig. 6 Overview of the jamming gripper at the end of Process 7 explained in Sect. 2.6

Fig. 6 shows the state of the jamming gripper at the end of Process 7 explained in Sect. 2.6. Following depressurization, the granular assembly becomes sufficiently constrained to maintain the deformed configuration of the gripper. As a result, the gripper remains in close contact with the target object, effectively sustaining the grasp. Although the target object has slightly lowered compared with the state shown in Fig. 5, c, it subsequently maintains this height, indicating a stable grasp. Although the microscopic jamming state is not quantified in the present study, the resulting mechanical stabilization reproduces the essential macroscopic behavior required for grasping.

4. Discussions and Conclusions

In this study, we presented a simulation method for reproducing the behavior of a jamming gripper, providing details of the software and implementation.

The proposed method employs the discrete element method (DEM) simulator LAMMPS to reproduce the grasping mechanism based on the jamming transition. Elements not typically addressed in DEM, such as the elastic membrane, target object, and depressurization operation, were also implemented. For the membrane and target object, particles were arranged and bonded according to custom three-dimensional shapes, and the interaction parameters were defined accordingly. The depressurization operation was simulated by reading particle configurations, obtaining particle positions, and applying a pseudo-impulse function.

The simulated gripper exhibited behavior consistent with that of a real jamming gripper, confirming that the proposed method can effectively reproduce its operation. Future work includes implementing suction forces due to pressure differences and reproducing jamming transitions through mechanisms other than depressurization, such as those observed with magnetic fluids.

Acknowledgments

This work is partially supported by the Grant-in-Aid of MEXT for Scientific Research (Grant No. JP24K06974, No. JP24K07193, and No. JP25K01063).

References

1. **Shintake, J.; Cacucciolo, V.; Floreano, D.; Shea, H.** 2018. Soft Robotic Grippers, Advanced Materials 30(29): 1707035. https://doi.org/10.1002/adma.201707035.
2. **Brown, E.; Rodenberg, N.; Amend, J.; Mozeika, A.; Steltz, E.; Zakin, M.R.; Lipson, H.; Jaeger, H.M.** 2010. Universal robotic gripper based on the jamming of granular material, Proceedings of the National Academy of Sciences (PNAS) 107(44): 18809-18814. https://doi.org/10.1073/pnas.1003250107.
3. **Koivikko, A.; Drotlef, D.M.; Sitti, M.; Sariola, V.** 2021. Magnetically switchable soft suction grippers, Extreme Mechanics Letters 44: 101263. https://doi.org/10.1016/j.eml.2021.101263.
4. **Licht, S.; Collins, E.; Mendes, M.L.; Baxter, C.** 2017. Stronger at Depth: Jamming Grippers as Deep Sea Sampling Tools, Soft Robotics 4(4): 305-316. https://doi.org/10.1089/soro.2017.0028.
5. **Tadakuma, K.; Fujimoto, T.; Watanabe, M.; Shimizu, T.; Takane, E.; Konyo, M.; Tadokoro, S.** 2020. Fire resistance deformable soft gripper based on wire jamming mechanism, 2020 3rd IEEE International Conference on Soft Robotics (RoboSoft): 740-747. https://doi.org/10.1109/RoboSoft48309.2020.9116036.
6. **Zhakypov, Z.; Heremans, F.; Billard, A.; Paik, J.** 2018. An Origami-Inspired Reconfigurable Suction Gripper for Picking Objects With Variable Shape and Size, IEEE Robotics and Automation Letters 3(4): 2894-2901. https://doi.org/10.1109/LRA.2018.2847403.
7. **Mantriota, G.** 2007. Optimal grasp of vacuum grippers with multiple suction cups, Mechanism and Machine Theory 42(1): 18-33. https://doi.org/10.1016/j.mechmachtheory.2006.02.007.
8. **Cao, C.; Gao, X.; Conn, A.T.** 2019. A Magnetically Coupled Dielectric Elastomer Pump for Soft Robotics, Advanced Materials Technologies 4(8): 1900128. https://doi.org/10.1002/admt.201900128.
9. **Hao, Y.; Biswas, S.; Hawkes, E.W.; Wang, T.; Zhu, M.; Wen, L.; Visell, Y.** 2021. A Multimodal, Enveloping Soft Gripper: Shape Conformation, Bioinspired Adhesion, and Expansion-Driven Suction, IEEE Transactions on Robotics 37(2): 350-362. https://doi.org/10.1109/TRO.2020.3021427.
10. **Gunderman, A.L.; Collins, J.A.; Myers, A.L.; Threlfall, R. T.; Chen, Y.** 2022. Tendon-Driven Soft Robotic Gripper for Blackberry Harvesting, IEEE Robotics and Automation Letters 7(2): 2652-2659. https://doi.org/10.1109/LRA.2022.3143891.
11. **Behringer, R. P.** 2015. Jamming in granular materials, Comptes Rendus. Physique 16(1): 10-25. https://doi.org/10.1016/j.crhy.2015.02.001.
12. **Amend, J.R.; Brown, E.; Rodenberg, N.; Jaeger, H.M.; Lipson, H.** 2012. A Positive Pressure Universal Gripper Based on the Jamming of Granular Material, IEEE Transactions on Robotics 28(2): 341-350. https://doi.org/10.1109/TRO.2011.2171093.
13. **Santarossa, A.; D'Angelo, O.; Sack, A.; Pöschel, T.** 2023. Effect of particle size on the suction mechanism in granular grippers, Granular Matter 25: 16. https://doi.org/10.1007/s10035-022-01306-7.
14. **Fitzgerald, S.G.; Delaney, G.W; Howard, D.; Maire, F.** 2021. Evolving soft robotic jamming grippers, Proceedings of the Genetic and Evolutionary Computation Conference (GECCO '21): 102-110. https://doi.org/10.1145/3449639.3459331.
15. **Götz, H.; Santarossa, A.; Sack, A.; Pöschel, T; Müller, P.** 2021. Soft particles reinforce robotic grippers: robotic grippers based on granular jamming of soft particles, Granular Matter 24: 31. https://doi.org/10.1007/s10035-021-01193-4.

16. **Santarossa, A.; Pöschel, T.** 2024. Enhanced interlocking in granular jamming grippers through hard and soft particle mixtures, Granular Matter 26: 105. https://doi.org/10.1007/s10035-024-01475-7.
17. **Santarossa, A.; D'Angelo, O.; Sack, A.; Pöschel, T.** 2025. All-terrain granular gripper, Particuology 104: 283-288. https://doi.org/10.1016/j.partic.2025.07.006.
18. **Thompson, A.P.; Aktulga, H.M.; Berger, R.; Bolintineanu, D.S.; Brown, W.M.; Crozier, P.S.; in 't Veld, P.J.; Kohlmeyer, A.; Moore, S.G.; Nguyen, T.D.; Shan, R.; Stevens, M.J.; Tranchida, J.; Trott, C.; Plimpton, S.J.** 2022. LAMMPS - a flexible simulation tool for particle-based materials modeling at the atomic, meso, and continuum scales, Computer Physics Communications 271: 108171. https://doi.org/10.1016/j.cpc.2021.108171.
19. **Cundall, P.A.; Strack, O.D.L.** 1979. A discrete numerical model for granular assemblies, Géotechnique 29(1): 47-65. https://doi.org/10.1680/geot.1979.29.1.47.
20. **Pöschel, T.; Schwager, T.** 2005. Computational Granular Dynamics: Models and Algorithms. Berlin, Heidelberg: Springer. 322p. https://doi.org/10.1007/3-540-27720-X.
21. **Plimpton, S.** 1995. Fast Parallel Algorithms for Short-Range Molecular Dynamics, Journal of Computational Physics 117(1): 1-19. https://doi.org/10.1006/jcph.1995.1039.
22. **Hertz, H.** 1882. Über die berührung fester elastischer körper, Journal für die reine und angewandte Mathematik 92: 156-171. https://doi.org/10.1515/crll.1882.92.156.
23. Blender Foundation. 2026. Blender. Available at: https://www.blender.org.

S. Hokada, S. Takada

NUMERICAL IMPLEMENTATION OF A JAMMING GRIPPER VIA DISCRETE ELEMENT METHOD IN LAMMPS

S u m m a r y

In this study, we propose a discrete element method (DEM)-based simulation framework for reproducing the behavior of a jamming gripper using the open-source simulator LAMMPS. The granular medium is modeled explicitly by particle-particle interactions, while key components not natively supported in DEM, namely the elastic membrane, depressurization process, and rigid structural parts, are implemented using bonded particle networks, particle-based rigid bodies, and externally applied pressure-induced forces. Importantly, the entire model is realized solely through LAMMPS input scripts, without modifying source code or introducing custom packages. Numerical simulations demonstrate that the proposed model successfully reproduces the deformation of the membrane, the compression and mechanical stabilization of the granular assembly under depressurization, and stable grasping of a target object. The results confirm that the essential mechanisms of jamming grippers can be captured within a standard DEM framework.

Keywords: granular materials, jamming, jamming gripper, LAMMPS.

Received January 10, 2026
Accepted October 15, 2026



This article is an Open Access article distributed under the terms and conditions of the Creative Commons Attribution 4.0 (CC BY 4.0) License (http://creativecommons.org/licenses/by/4.0/).