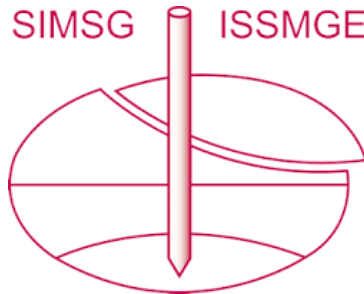


INTERNATIONAL SOCIETY FOR SOIL MECHANICS AND GEOTECHNICAL ENGINEERING



This paper was downloaded from the Online Library of the International Society for Soil Mechanics and Geotechnical Engineering (ISSMGE). The library is available here:

<https://www.issmge.org/publications/online-library>

This is an open-access database that archives thousands of papers published under the Auspices of the ISSMGE and maintained by the Innovation and Development Committee of ISSMGE.

The paper was published in the proceedings of the 10th European Conference on Numerical Methods in Geotechnical Engineering and was edited by Lidija Zdravkovic, Stavroula Kontoe, Aikaterini Tsiampousi and David Taborda. The conference was held from June 26th to June 28th 2023 at the Imperial College London, United Kingdom.

To see the complete list of papers in the proceedings visit the link below:

<https://issmge.org/files/NUMGE2023-Preface.pdf>

tiSPHi: A parallelised GPU-accelerated SPH framework for modelling robot-ground interaction

Zhibin Lei, Raul Fuentes

Institute of Geomechanics and Underground Technology, RWTH Aachen University, Aachen, Germany

ABSTRACT: Efficient locomotion of robots in real environments involves highly complex robot-ground interaction (RGI) problems. Granular geomaterials exhibit complex multiphase phenomena under dynamic impact, such as solid-like and fluid-like characteristics, together with an ejected gas-like phase. These phenomena occur in diverse situations, ranging from impact and penetration problems to locomotion over natural or man-made terrains. This range of behaviours is yet difficult to simulate accurately and efficiently. Here, we develop tiSPHi, a continuum model based on smoothed particle hydrodynamics (SPH) using Taichi parallel computing language. Using a simple Drucker-Prager model of geomaterials, we demonstrate a computationally efficient rapid modelling of granular locomotion of arbitrary intruders. Several tests are carried out to examine the capability of this model. Results show that this model provides a good computational efficiency with a balance of development productivity compared to other implementations, whilst, outperforming in terms of accuracy.

Keywords: Smoothed particle hydrodynamics (SPH); robot-ground interaction (RGI); GPU acceleration; tiSPHi

1 INTRODUCTION

Robots that can move efficiently in real environments are one of the key factors for human exploration of nature, including extraterrestrial environment exploration (Ding et al. 2022). So far, no robots have indicated the same level of efficiency and dynamism as animals while operating in natural environments. The interaction between robot mechanics, ground mechanics, and robot control and sensing is highly complex (Yang et al. 2018). Therefore, it is essential to conduct a comprehensive study of the robot-ground interaction (RGI) problem to simulate and develop robots that can effectively navigate non-uniform and unstructured real-life terrain.

This kind of interaction (e.g. running legged robot on sandy surface) represents a complex granular intrusion near free-surfaces. Unlike fluids that can be solved with the Navier-Stokes equations, the coupled system of an intruder and geotechnical granular media can lead to intricate multiphase phenomena and force responses (Schiebel et al. 2020).

To accurately and effectively capture such complex responses in simulation, it is necessary to use appropriate numerical modelling methods. Among many approaches, the smoothed particle hydrodynamics (SPH) method can efficiently model the general behavior of granular flows with large strains and displacements due to its mesh-free nature. The SPH method can be employed to solve partial differential equations associated with the conservation laws of mass, momentum, and energy. This is achieved by spatially discretising the equations into a set of particles, each

possessing material properties, that interact with each other through kernel functions with compact support domains. After introducing the elastoplastic constitutive model into SPH (Bui et al. 2008), the numerical results of granular material flows with large deformations have exhibited good agreement with experimental data (Chalk et al. 2020, Bui and Nguyen 2021). Meanwhile, the modelling of interactions between granular flows and rigid or flexible bodies through SPH frame has been shown to be a promising approach (Hu et al. 2021).

Furthermore, in modelling RGI problems, the scale of the intruder bodies being considered is relatively small compared to the overall size of the simulated domain, requiring high-resolution particle discretization. This can be computationally expensive and puts high demands on program efficiency. Recently, three effective methods have been summarised (Xu et al. 2023) to apply SPH to large-scale or multiscale applications on affordable hardware within a reasonable amount of time. Among these, GPU (Graphics Processing Unit) parallel computing has emerged as a new architecture for large-scale scientific computing, and using CUDA (Compute Unified Device Architecture) can serve as the core foundation for implementing the entire numerical method.

There are various open-source software packages available, such as DualSPHysics (Crespo et al. 2015), SPLisHSPlasH (Koschier et al. 2019), Chrono (Hu et al. 2021), and SPHinXsys (Zhang et al. 2021), which can efficiently solve SPH problems. However, these packages primarily focus on simulating fluids or elastic bodies. For geomaterials that exhibit elastic-plastic

properties, there are programs such as LOQUAT (Peng et al. 2019) and Stress-Particle SPH (Chalk et al. 2020), but they suffer from limitations of low efficiency or lack of portability. In addition, all the above programs are implemented using lower-level languages, such as C++, FORTRAN, CUDA, which limits their versatility. It is also challenging to achieve a high level of optimization for these packages. The PySPH (Ramachandran et al. 2021) is written in the more accessible and popular Python language. However, its functionality is not sufficient to solve RGI problems, and the development of new cases still presents several challenges.

Recent research in domain-specific languages has effectively bridged the gap between productivity and performance and Taichi Language (Hu 2021) is one of the most notable languages in this field. Taichi is an open-source parallel computing language embedded in Python and provides both good flexibility and optimisation for data structures and algorithms. It uses just-in-time frameworks to offload the Python source code to native GPU or CPU instructions, offering the performance at both development time and runtime and providing good portability. Through benchmark comparisons with highly optimised CUDA code, Taichi has been shown to achieve performance comparable in most cases or better in some complex problems.

Therefore, we have developed a new SPH framework, tiSPHi, using the Taichi Language. The main purpose is to accurately and efficiently simulate the complex dynamics of particle intrusion, and to promote the research and application of SPH in solving RGI problems. In the remaining part of the paper, the theories and the approaches of SPH are detailed, followed by the code structure and implementation. Afterwards, we present several numerical examples that showcase the solver's accuracy, efficiency, and stability.

2 METHODOLOGY

2.1 Governing equations

Given the interest of this work in continuum modelling of ground material, granular flows such as sand are considered to be continuous media flows. Considering the interaction between ground materials and moving rigid bodies, the entire computational domain of the system can be represented as $\Omega = \Omega_F + \Omega_B$, where Ω_F and Ω_B are the sub-domains occupied by the granular flows and rigid bodies, respectively.

In continuum mechanics, the conservations of mass and momentum for a granular material yield the following differential equations in a Lagrangian frame, respectively:

$$\frac{d\rho}{dt} = -\rho \nabla \cdot \mathbf{v} \quad (1)$$

$$\frac{d\mathbf{v}}{dt} = \frac{1}{\rho} \nabla \cdot \boldsymbol{\sigma} + \mathbf{f}^{ext} \quad (2)$$

where ρ is the density, \mathbf{v} is the velocity, $\boldsymbol{\sigma}$ is the total stress tensor (which is negative for compression), and \mathbf{f}^{ext} is the external body force such as gravity.

To account for the mechanical properties of the particulate material and formulate a complete system of equations necessary for solving the boundary value problem, we adopt an elastoplastic constitutive model that utilizes the Drucker-Prager yield criterion to determine the stress increment during the deformation of granular materials. A detailed description can be found in Bui and Nguyen (2021).

Incorporating the particle discretisation of SPH, we employ shape-matching constraints (Müller et al. 2005) to maintain the solid particle configurations. The initial particle placement is accomplished through solid voxelisation, and the constraint step during the simulation automatically respects the inertial properties of the object. As a result, a simple rigid body simulator is obtained, which is capable to handle solid boundaries with arbitrary geometry, whether static or in motion.

2.2 SPH discretisation and approaches

In the SPH method, the partial differential equations are spatially discretized using particles that advect with the flow field. A generic field variable $f(\mathbf{x})$ at position \mathbf{x}_i , also written as f_i , can be approximated using a finite set of sampling points \mathbf{x}_j located within $\|\mathbf{x}_i - \mathbf{x}_j\| < h$ as:

$$f_i = \sum_j f_j V_j W(\mathbf{x}_i - \mathbf{x}_j, h) \quad (3)$$

where V_j is the volume carried at \mathbf{x}_j , and $W(\mathbf{x}_i - \mathbf{x}_j, h)$, also written as W_{ij} , denotes the kernel function with support radius h . The kernel function chosen here is the Wendland C^2 kernel function. Bui and Nguyen (2021) have demonstrated that this function has a non-negative Fourier transform, suggesting that it can address the pairing instability issue.

The derivatives of $f(\mathbf{x})$ at the location of SPH particle \mathbf{x}_i can be computed by using the derivatives of kernel function in the interpolation as:

$$\nabla f_i = \sum_j (f_j - f_i) V_j \nabla_i W_{ij} \quad (4)$$

Hence, the consistent SPH discretization of the governing equations is given as:

$$\frac{d\rho_i}{dt} = -\rho_i \sum_j (v_j^\alpha - v_i^\alpha) V_j \nabla_i^\alpha W_{ij} \quad (5)$$

$$\frac{dv_i^\alpha}{dt} = \sum_j \rho_j \left(\frac{\sigma_j^{\alpha\beta}}{\rho_j^2} + \frac{\sigma_i^{\alpha\beta}}{\rho_i^2} \right) V_j \nabla_i^\beta W_{ij} + f_i^\alpha \quad (6)$$

In SPH, searching for the neighbours of every particle is a critical step and can be computationally intensive that dominates the whole simulating time. We therefore conduct an efficient prefix sum algorithm and counting sort to speed up the neighbour search process.

To solve the problem of kernel truncation of particles near the boundary, we use layers of dummy particle method which is suitable for soil dynamic problems (Bui et al. 2008) for all boundaries. Moreover, the smoothed velocity field and stress field of the Ω_F are extrapolated to the dummy particle positions (Adami et al. 2012, Feng et al. 2021) to impose the no-slip boundary condition.

As for the artificial terms in SPH, we used the traditional artificial viscosity and linear viscous damping model (Nguyen et al. 2017) to damp out the unwanted velocity or stress oscillations. Meanwhile, in order to remove the short-length-scale noise in granular flows under large shear deformation, stress regularisation (Nguyen et al. 2017) using Shepard filter is adopted.

Three explicit schemes have been utilised to update the governing equations in time, which are first-order Symplectic-Euler scheme (SE), second-order Leap-Frog scheme (LF) (Bui et al. 2008) and fourth-order Runge-Kutta scheme (RK) (Chalk et al. 2020). The size of Δt is constrained by the Courant-Friedrichs-Levy (CFL) condition. After conducting several tests, the LF scheme is selected in this paper as it yields more accurate and stable results than SE, while maintains less computing time and lower memory usage than RK.

Finally, the particle's location is advected using the XSPH technique, which can reduce particle disordering.

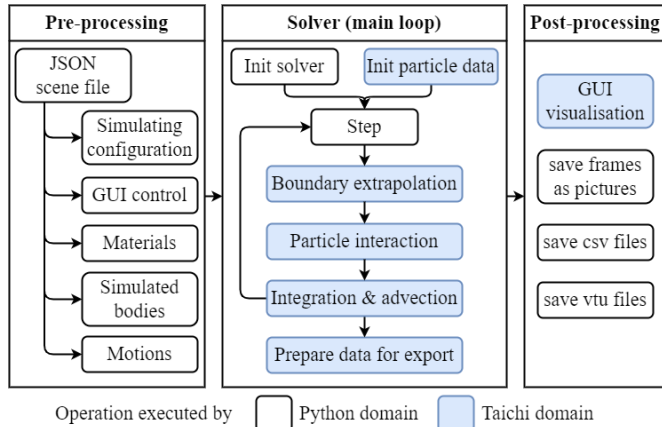


Figure 1. Flowchart of the simulation process in tiSPHi.

2.3 Implementation

Figure 1 depicts the detailed computational procedure of tiSPHi. The Python domain works as an execution manager. It is used for loading and saving data, initialising the settings of simulation, and linking the Taichi kernels in a predetermined order. The Taichi domain executes all the actual computations that are written in Taichi computing kernels. Components utilising different methods are abstracted to be independent in this framework, so it is easy to add new features. In addition, GPU

device can also be used for real-time rendering of the simulation process, which provides great convenience for program development. A noteworthy point is that the entire code for the whole process in the current model is less than 3,000 lines.

3 VALIDATION

In this section, we adopt the versatility of tiSPHi through several test cases, including hydrostatic test, granular column collapse, and dynamic plate intrusions. To facilitate clarity, all simulations are performed under two-dimensional plane strain conditions and with simple geometries. The detailed parameters of the models can be found in Table 1 and will not be repeated below.

The host CPU used for the computations in this study is the Intel Core i7-7700K, which has 8 computing cores and a base speed of 4.2GHz. Moreover, the GPU device used is the NVIDIA GeForce RTX3060 Laptop, which has 3840 CUDA cores with a 1320MHz base speed.

Table 1. Adopted parameters in case study.

Parameters	Unit	Adopted values for test cases		
		Granular material	Horizontal intrusion	Vertical intrusion
Density	kg/m ³	2040	900	2500
Young's modulus	MPa	5.84	2	10
Poisson's ratio	-	0.3	0.3	0.3
Friction angle	deg	21.9	30	31.8

3.1 Hydrostatic test

Firstly, we simulate a rectangular container filled with granular material to a depth of $H = 0.1\text{m}$. The container has a size of $W = 0.2\text{m}$ and $H = 0.12\text{m}$. Figure 2(a) displays the initial setup, where grey particles represent the wall boundary and brown particles denote the granular phase. The particle resolution is $\Delta x = 0.002\text{m}$, resulting in a total of 5k particles.

The simulation is carried out for 2 seconds of physical time (which takes around 11min to compute over 181k steps on GPU), and the particles settle down to a steady stress field, as depicted in Figure 2(b). It exhibits a satisfactory linear-in-depth stress profile. Additionally, no non-physical motion, such as a separation of particles from the walls at the free surface, is observed. They both indicate the correctness of the implemented boundary condition and constitutive model.

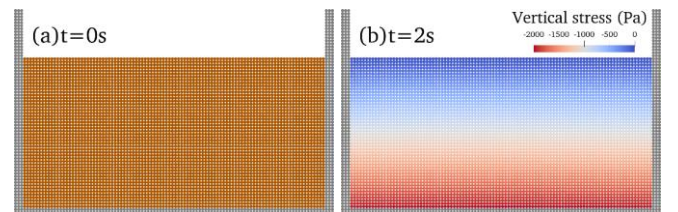


Figure 2. Initial particle positions at $t = 0\text{s}$ (a) and the vertical stress field at $t = 2\text{s}$ (b).

Figure 3 illustrates the vertical stress carried by the particle located at the center of the box over time. It is evident that the oscillation of the vertical stress induced by the abrupt application of gravity is effectively eliminated, and the exact value is eventually achieved.

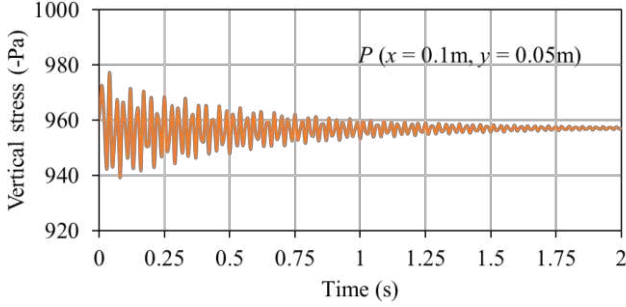


Figure 3. Temporal vertical stress profile of the particle at the central position.

3.2 Granular column collapse

To assess the precision of the numerical scheme in simulating large deformation problems, a 2D granular column collapse test is performed and compared with the experimental results presented by Nguyen et al. (2017). The granular column has dimensions of 0.2m in length and 0.1m in height, with the walls modelled as no-slip boundaries. A total of 5k particles are generated with an initial particle spacing of $\Delta x = 0.002\text{m}$, and it takes 5min to run 0.8s physical seconds on the GPU device.

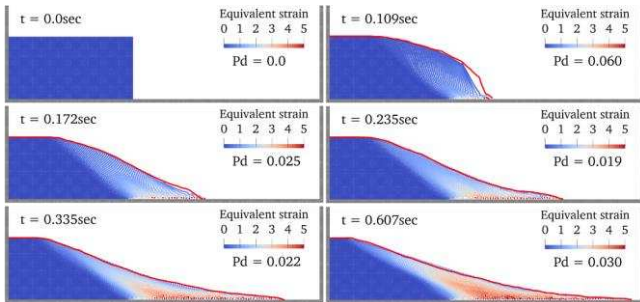


Figure 4. Comparison between the numerical simulation and the experiment for 2D granular column collapse progress.

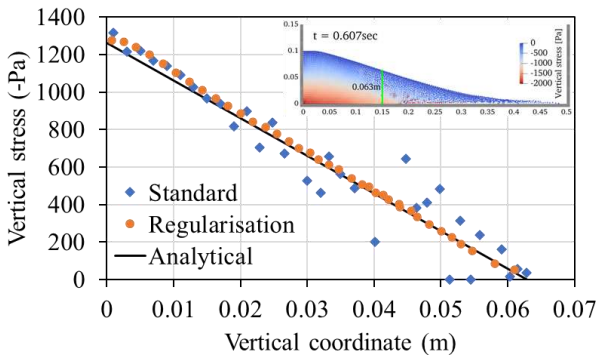


Figure 5. Vertical stress profile of the final deposit at $x = 0.15\text{m}$. The inset shows the vertical stress distribution of the standard SPH approach without stress regularisation.

Figure 4 displays the equivalent strain profiles of the granular flows obtained from the simulation at several

time intervals, along with the corresponding free-surface lines from the experimental results plotted in solid red lines for comparison. To quantify the deviation between the numerical and experimental results, we calculate the phase difference coefficient P_d from Gomez-Gesteira et al. (2010), where a value of $P_d \rightarrow 0$ indicates a perfect agreement. It can be seen that the SPH models predict fairly well the time evolution of surface deformation and final run-out distance, with the phase difference P_d generally on the order of 1%.

The predicted vertical stresses at $x = 0.15\text{m}$ are plotted in Figure 5 against the theoretical solution. It is apparent that the standard simulation without stress regularisation provides an accurate and smooth stress profile within the undisturbed region (below 0.02m), but the stress distribution is scattered in the highly deformed region. After applying stress regularisation, the spurious stress profile is improved, resulting in a smooth and accurate prediction of stress can be achieved for both undisturbed and deformed region.

Based on the above two validations, it can be concluded that the SPH approach employed in this study is effective in accurately predicting the behaviour of granular materials.

3.3 Dynamic intrusion

Complex dynamic intrusions near the free surface of granular materials represent a typical form of RGI problem. To assess the accuracy of our SPH approach in modelling intrusions, we studied two converse cases of forced motion of plates. Then the results were evaluated based on the study conducted by Agarwal et al. (2021).

3.3.1 Submerged horizontal intrusion

In this problem, a thin plate is submerged in a granular material at a certain depth and is dragged horizontally at constant speeds. The length of the plate is 0.016m, and the depth from the bottom of the plate to the ground surface is 0.03m. For these cases with 400k particles, simulating 0.1 seconds of physical time on the GPU takes approximately 3.6h.

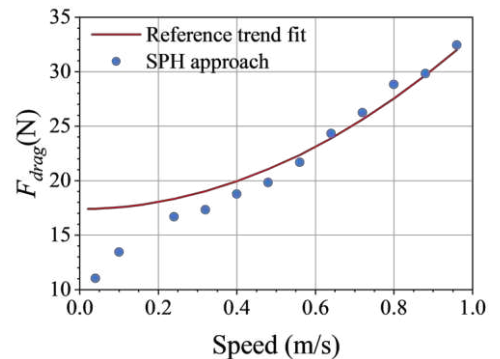


Figure 6. Comparison of the drag force data obtained from horizontal intrusion simulations (blue circles) and the trend fit (red solid line).

The observed drag force distributions obtained from SPH simulations at varying intruding speeds are plotted in Figure 6 as filled circles, with each representing the drag force value when the plate moved a certain distance at the corresponding velocity. The fitted line is calculated using the trend $F_{drag} = K|z| + \lambda\rho Av^2$ (Schiebel et al. 2020), where K and λ are constants, $|z|$ is the depth, ρ is the effective granular density, A is the area of plate which denotes length in 2D, and v is the horizontal plate velocity. It can be seen that predictions from our SPH modelling match the trend well for the reference constant value $K = 580\text{N/m}$ and $\lambda = 1.1$ as the speed increases, although there appears more than 20% deviations when the speed is very low.

A comprehensive understanding and comparison of the trends in the drag force can be obtained by examining the simulation results. Figure 7 indicates the deformation profiles around the plate at two selected velocities, which are about one order of magnitude apart. The profiles suggest that most of the drag forces come from the relatively stable Coulomb-wedge structure ahead of the intruding plates. The intruder traction arises from pressing the granular material in front of the plate upward and to the right, towards the free-surface. And the most notable difference in material motion occurs in the rear flow zone. This zone is either in the separated phase under high-speed intrusion or newly consolidated after materials fall and fill in the gap behind the slowly moving plate. These simulations demonstrate the same particle motion phenomenon and force response in horizontal plate intrusion as the reference study.

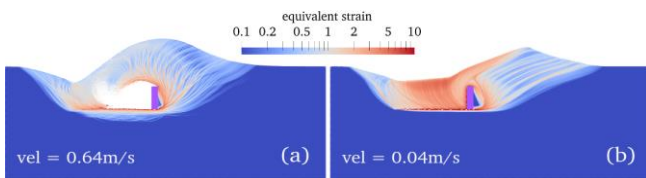


Figure 7. Modelling for horizontal plate intrusions at both high and low speeds. Coloured by the equivalent strain.

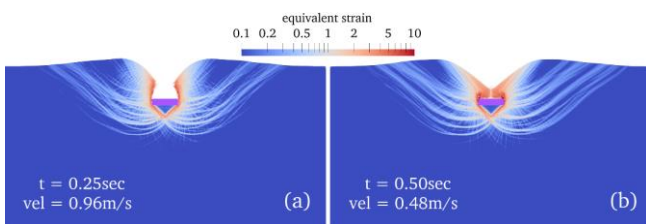


Figure 8. Modelling for vertical plate intrusions at both high and low speeds. Coloured by the equivalent strain.

3.3.2 Symmetric vertical intrusion

In contrast to the horizontal intrusion, the thin plate also intrudes vertically into the granular materials at different speeds (0.96m/s and 0.48m/s). As shown in Figure 8, two independent bulk flow regions, or Coulomb-wedge type structures, are formed ahead of the plate during the vertical intrusion and maintain their shape. Moreover, two non-interacting bulk flow regions are also formed

on either side of the plate intrusion. It again indicates that the small contribution of resistive force response from the dynamic structural correction, similar to the horizontal plate intrusions. Thus, the motions and geometries of material flow generated by our approach agree well with the reference study.

3.4 Performance analysis: water dambreak

We also implemented weakly compressible SPH model without any optimisation in tiSPHi to evaluate our computational efficiency by simulating a classic dambreak model of the same size using several other SPH fluid solvers. These software packages include highly optimised PySPH, SPLisHSPlasH, SPHinXsys, as well as a simple MATLAB serial program. A 5.366m length water tank is initially covered with a $2 \times 1\text{m}$ rectangular size of water on the lower left side and is discretized into 5k water particles with a resolution of $\Delta x = 0.02\text{m}$.

Figure 9 shows the computation time required to perform 10k time steps for various software on different devices, as well as the times needed to compute 1 second of physical time. The latter times were calculated based on the physical times reached by the former simulations (0.6s, 0.6s, 0.831s, 0.831s, 22.169s, 11.732s, 0.01s, respectively) extrapolated to 1s (e.g., $38.3\text{s} = 23\text{s}/0.6$).

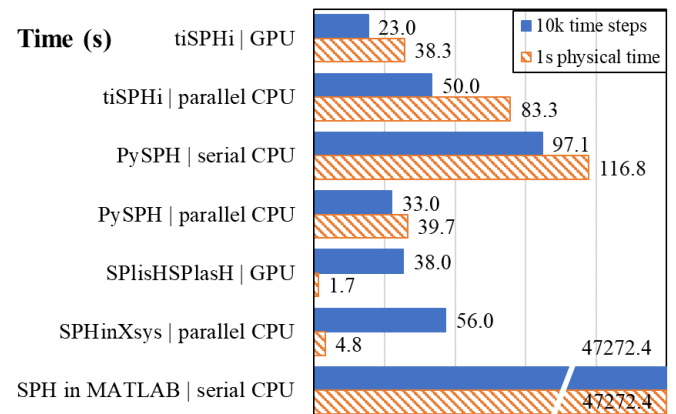


Figure 9. Comparison of the time (in seconds) required to simulate the water dambreak using different approaches and devices. The shorter bars indicate higher performance.

The results clearly indicate that the efficiency of the MATLAB approach is unacceptably slow, while PySPH performs well on both serial and parallel CPU device thanks to its in-depth optimisation. Because of the ability to use very large variable time steps, SPLisHSPlasH and SPHinXsys only require very short times to run 1 physical second. From the perspective of time steps, tiSPHi, without any manual optimisation, has almost reached or even surpassed the computational efficiency of existing programs when running on GPU or parallel CPU devices. However, from the perspective of physical time, there is still a gap between tiSPHi and existing highly optimised programs due to the limitation of the time step. This is mainly because the time step of the current

framework is fixed and relatively small owing to the CFL condition, especially when the particle discretisation spacing is very small.

4 CONCLUSIONS AND FUTURE WORK

Based on Taichi Language, we have developed a new, general parallel SPH framework, tiSPHi, to simulate interactions between robots and granular geomaterials. The framework has been designed with high modularity in the architecture and functionality, allowing for code readability and computational efficiency, while also allowing flexibly for future development. Results from various tests show that it provides convergence and accurate predictions of deformation, stress distribution, and force response, demonstrating the accuracy, robustness, and applicability of this framework.

Future development of tiSPHi will concentrate on improving the performance, accuracy, and functionality. Firstly, there are two ways to further enhance the efficiency: (1) improving the algorithm through adaptive particle resolution and adaptive time step, and (2) reducing memory usage by improving the numerical precision of single-precision floating-point data and adopting more efficient neighbourhood search methods. Secondly, the constitutive model and coupling system will be updated. Finally, addressing the RGI problem will require modelling more complex robot motions, such as rotation and deformation.

5 ACKNOWLEDGEMENTS

The support of China Scholarship Council is gratefully acknowledged.

REFERENCES

- Adami, S., Hu, X.Y., and Adams, N.A. 2012. A generalized wall boundary condition for smoothed particle hydrodynamics. *Journal of Computational Physics* **231**(21), 7057–7075.
- Agarwal, S., Karsai, A., Goldman, D.I., et al. 2021. Surprising simplicity in the modeling of dynamic granular intrusion. *Science Advances* **7**(17), eabe0631.
- Bui, H.H., Fukagawa, R., Sako, K., et al. 2008. Lagrangian meshfree particles method (SPH) for large deformation and failure flows of geomaterial using elastic-plastic soil constitutive model. *International Journal for Numerical and Analytical Methods in Geomechanics* **32**(12), 1537–1570.
- Bui, H.H. and Nguyen, G.D. 2021. Smoothed particle hydrodynamics (SPH) and its applications in geomechanics: From solid fracture to granular behaviour and multiphase flows in porous media. *Computers and Geotechnics* **138**, 104315.
- Chalk, C.M., Pastor, M., Peakall, J., et al. 2020. Stress-Particle Smoothed Particle Hydrodynamics: An application to the failure and post-failure behaviour of slopes. *Computer Methods in Applied Mechanics and Engineering* **366**, 113034.
- Crespo, A.J.C., Domínguez, J.M., Rogers, B.D., et al. 2015. DualSPHysics: Open-source parallel CFD solver based on Smoothed Particle Hydrodynamics (SPH). *Computer Physics Communications* **187**, 204–216.
- Ding, L., Zhou, R., Yuan, Y., et al. 2022. A 2-year locomotive exploration and scientific investigation of the lunar farside by the Yutu-2 rover. *Science Robotics* **7**(62), eabj6660.
- Feng, R., Fourtakas, G., Rogers, B.D., et al. 2021. Large deformation analysis of granular materials with stabilized and noise-free stress treatment in smoothed particle hydrodynamics (SPH). *Computers and Geotechnics* **138**, 104356.
- Gomez-Gesteira, M., Rogers, B.D., Dalrymple, R.A., et al. 2010. State-of-the-art of classical SPH for free-surface flows. *Journal of Hydraulic Research* **48**(sup1), 6–27.
- Hu, W., Rakhsha, M., Yang, L., et al. 2021. Modeling granular material dynamics and its two-way coupling with moving solid bodies using a continuum representation and the SPH method. *Computer Methods in Applied Mechanics and Engineering* **385**, 114022.
- Hu, Y. 2021. The Taichi High-Performance and Differentiable Programming Language for Sparse and Quantized Visual Computing. Massachusetts Institute of Technology, Boston.
- Koschier, D., Bender, J., Solenthaler, B., et al. 2019. Smoothed Particle Hydrodynamics Techniques for the Physics Based Simulation of Fluids and Solids. *Eurographics 2019 - Tutorials*.
- Müller, M., Heidelberger, B., Teschner, M., et al. 2005. Meshless deformations based on shape matching. *ACM Transactions on Graphics* **24**(3), 471–478.
- Nguyen, C.T., Nguyen, C.T., Bui, H.H., et al. 2017. A new SPH-based approach to simulation of granular flows using viscous damping and stress regularisation. *Landslides* **14**(1), 69–81.
- Peng, C., Wang, S., Wu, W., et al. 2019. LOQUAT: an open-source GPU-accelerated SPH solver for geotechnical modeling. *Acta Geotechnica* **14**(5), 1269–1287.
- Ramachandran, P., Bhosale, A., Puri, K., et al. 2021. PySPH: A Python-based Framework for Smoothed Particle Hydrodynamics. *ACM Transactions on Mathematical Software* **47**(4), 34:1-34:38.
- Schiebel, P.E., Astley, H.C., Rieser, J.M., et al. 2020. Mitigating memory effects during undulatory locomotion on hysteretic materials. *eLife* **9**, e51412.
- Xu, F., Wang, J., Yang, Y., et al. 2023. On methodology and application of smoothed particle hydrodynamics in fluid, solid and biomechanics. *Acta Mechanica Sinica* **39**(2), 722185.
- Yang, G.-Z., Bellingham, J., Dupont, P.E., et al. 2018. The grand challenges of Science Robotics. *Science Robotics* **3**(14), eaar7650.
- Zhang, C., Rezavand, M., Zhu, Y., et al. 2021. SPHinXsys: an open-source multi-physics and multi-resolution library based on smoothed particle hydrodynamics. *Computer Physics Communications* **267**, 108066.