ARMA 24–284

Predicting scour in unlined rock spillways: a coupled analytical-numerical modeling approach

Weidner, L., George, M.F., and Clohan, D.
BGC Engineering Inc., Golden, Colorado, United States

This paper was prepared for presentation at the 58th US Rock Mechanics/Geomechanics Symposium held in Golden, Colorado, USA, 23-26 June 2024. This paper was selected for presentation at the symposium by an ARMA Technical Program Committee based on a technical and critical review of the paper by a minimum of two technical reviewers. The material, as presented, does not necessarily reflect any position of ARMA, its officers, or members. Electronic reproduction, distribution, or storage of any part of this paper for commercial purposes without the written consent of ARMA is prohibited. Permission to reproduce in print is restricted to an abstract of not more than 200 words; illustrations may not be copied. The abstract must contain conspicuous acknowledgement of where and by whom the paper was presented.

ABSTRACT: Hydraulic plucking of rock blocks bounded by discontinuities is one of the primary modes of scour in unlined spillways. The methods used in the current state-of-practice either model a single block in considerable detail or introduce many simplifications to reduce model runtime and complexity. We present a method for simulating block scour at the spillway scale using an open-source discrete fracture network and block slicing program coupled with FLOW-3D. We built a custom implementation of limit equilibrium block theory for this task which enables eroded blocks to have any convex polyhedral shape. Preliminary results from this method highlight that block shapes introduce significant anisotropy to erosion behavior, which is not captured in other methods.

1. INTRODUCTION

Hydraulic plucking of rock blocks bounded by discontinuities is known to be one of the primary modes of scour in rock masses, resulting in channelization and head cutting. As such, understanding this process has immense implications for the design and maintenance of engineering structures such as spillways and dam foundations interacting with water. Modeling block scour can be accomplished using empirical relationships (Annandale, 1995), physics-based analytical stability solutions (Bollaert & Schleiss, 2002; George & Sitar, 2016; Goodman & Shi, 1985), or numerical simulations using coupled fluid-solid numerical methods (e.g., Gardner, 2023). Empirical methods are fast to apply at a spillway scale but are based on simplified inputs. Analytical methods apply directly to individual blocks but require many inputs of the forces acting on each block face - which can be time consuming to estimate.

Three-dimensional numerical methods currently in development can model scour and plucking of individual blocks directly by modelling fluid-solid interactions, but they are typically impractical at the spillway scale due to prohibitively large computational requirements (Gardner, 2023; Gardner & Sitar, 2019; Teng et al., 2023). Existing numerical models used in practice generally simplify the geometry to two-dimensional cross sections (Bollaert & Schleiss, 2005). Current methods are unable to both explicitly model hydraulic plucking mechanisms at the spillway scale and do this in three dimensions with arbitrary block shapes. Using the actual 3D block geometry is required to represent the block kinematics which is widely known to be a predominant factor in stability of rock blocks. Simplifications using rectangular blocks are unable to account for anisotropy caused by oblique joint orientations, and the stability of more irregular polyhedrons may vary significantly from a regular cube (George & Sitar, 2016).

This research proposes a coupled analytical-numerical approach as a method for spillway-scale scour modeling in 3D. Specifically, we propose to evaluate 3D block plucking using a novel implementation of the Block Theory, limit equilibrium stability approach originally developed by Goodman and Shi (1985) and then adapted to scour applications by George and Sitar (2016) and George (2023). This is then coupled with a 3D computational fluid dynamics (CFD) software (FLOW-3D) to estimate water flow conditions around the blocks.

2. METHODS

The method we propose is summarized in Fig. 1. First, a rock mass geometry set of interlocking blocks is generated using discontinuity data. Once the base model geometry is created, each block is analyzed to identify the subset of blocks which are kinematically removable (and unstable, for subsequent iterations when flow velocity data are given). This is discussed in more detail in Section 2.4. The geometry is then converted into a format amenable to the FLOW-3D CFD software (Flow Science Inc., 2023), which then computes flow velocities above the modeled rock surface. This creates a feedback loop in which exported velocity information is used to analytically compute block stability and update the model surface, which is then fed back into FLOW-3D. The following sections describe each of these steps in detail.

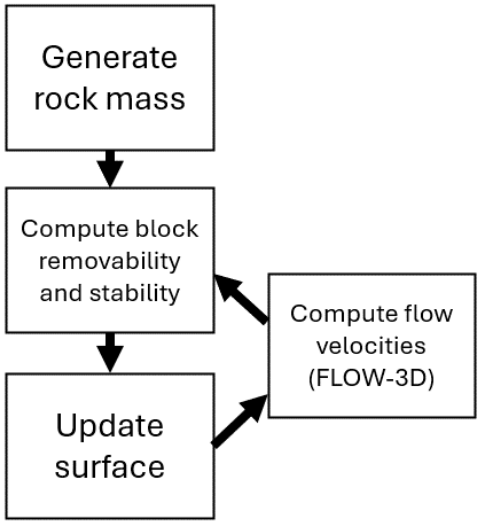

Fig. 1. Proposed modeling workflow for spillway rock scour.

2.1. *Block Model Generation*

Generating discrete fracture networks (DFNs) and systems of interacting rock blocks to model various geo-structural and hydrologic phenomena is challenging (Ivars et al., 2011). There are many open-source and commercial DFN generation tools (Hyman et al., 2015); however, most available codes for generating 3D blocky rock masses from DFNs are either commercial or otherwise restricted/unable to be adapted or modified (Rasmussen, 2020) and may be too inefficient for large-scale model development (Gardner et al., 2017). Rasmussen (2020) developed an open-source Python and C++ based library, *UnBlocks-gen* for generating rock masses based on the rock slicing method of Boon (2015) which we use in this paper.

UnBlocks-gen can generate a series of individual blocks within a rock mass based on any number of non-persistent deterministic and probabilistically generated joints. The blocks are first generated in a cubic model space, and then geometries can be cut away after, creating a ground surface. We used *UnBlocks-gen* to create a hypothetical east-facing emergency spillway with a uniform slope of 8 degrees (Fig. 2). For this example, blocks were generated from four probabilistic discontinuity sets with relatively arbitrary parameters. For a real case study, each of these parameters can be adjusted to match observed conditions. To control the number of discontinuities, a target density of fractures must be set, and the code continues to generate fractures until the density metric is met. We set the density based on the number of fractures in each set per unit volume (P_{30}), with values chosen of 10^{-4} to 10^{-5}, generating relatively large blocks through trial and error. Using a higher volume density may be desired based on rock mass characteristics and field data, but this increases the amount of time needed to generate the block model and perform subsequent block updating steps for each iteration. We estimate that the current implementation is reasonably efficient for up to a few tens of thousands of blocks. The orientation variance of each generated fracture from the mean is controlled by the von Mises-Fisher distribution concentration parameter, which was set to 50 for all sets. The Fisher distribution is a spherical probability distribution, used to statistically represent discontinuity orientation data. These parameters resulted in the generation of 6,108 individual blocks, which are saved as OBJ mesh files describing the vertices and faces.

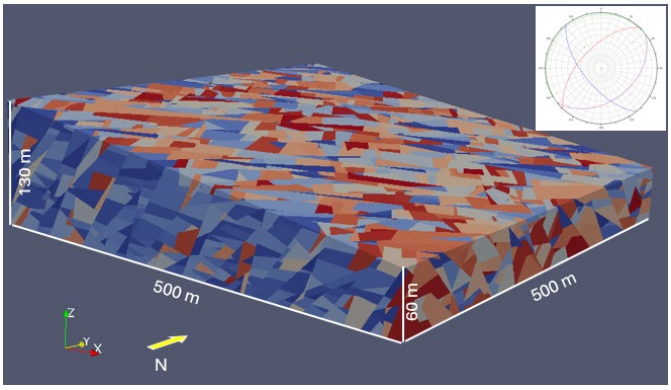

Fig. 2. Idealized spillway block model generated using *UnBlocks-gen*. Stereonet in the upper right shows the mean orientations of the four discontinuity sets. Block colors ranging from blue to red are randomly distributed to distinguish between adjacent blocks.

After blocks are exported from *UnBlocks-gen*, they are processed using a custom Python package developed by the authors which ingests blocks, computes useful descriptive information, and performs the block stability routines as described in later sections. All steps other than computing flow information in FLOW-3D are performed using this Python package. The Python package was developed using publicly available libraries, such as scipy and shapely. We also used CloudCompare Python bindings (cloudComPy) for a few operations.

We track each block using a custom data class, which records the vertices and faces of the block, face areas, plane equations and inward-facing normal vectors for each face, and whether each face is a "joint face" (within the rock mass) or a "surface face" (free, exposed to the surface) (Fig. 3).

Tracking which faces are exposed to the surface is an important problem because it determines which blocks are kinematically removable by hydraulic forces. The model is initialized by first computing the distance between all block faces and a base mesh representing the uneroded ground surface. Faces that are coplanar with the surface mesh are labeled as surface faces and faces not intersecting the surface are labeled "joint faces". As blocks are removed from the model by water, the removal algorithm performs a similar procedure to update which faces are newly exposed based on their coplanarity and intersection area with the faces of a block just removed.

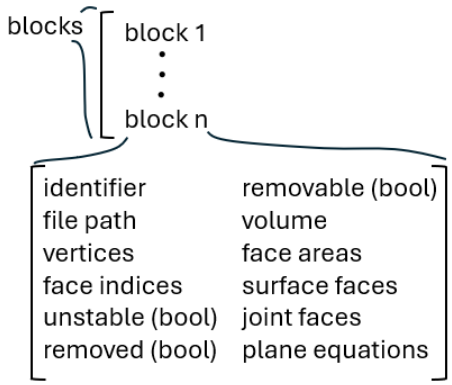

Fig. 3. Illustation of the block class data structure used to manage and update information for each block.

2.2. Transferring Block Geometry to FLOW-3D

The geometry created by a series of thousands of convex polyhedrons with arbitrary numbers of vertices per face is inherently complex, FLOW-3D does not natively handle thousands of individual objects. However, since only the model surface is needed to compute flow velocities, efficiency can be increased by only passing FLOW-3D the aggregate surface of the block model (note that internal joint hydraulic pressures are accounted for analytically during stability calculations by the dynamic pressure coefficients and not modeled numerically).

In general, creating an exterior surface mesh can be challenging to accomplish with high precision for rock masses with non-persistent joints because the mesh must be watertight. As blocks are removed, they expose portions of surfaces of underlying blocks. We have found that attempting to build the exterior model surface based on each individual block and block face results in computations which are either impractically slow (e.g., concave hull or alpha shapes) or are unstable due to limitations in geometric intersection tools (e.g., shapely, trimesh, libigl, Blender). For this work we use a robust, but brute-force method using cloudComPy, summarized as follows:

(i) Merge all blocks into a single non-watertight mesh object.

(ii) Sample points at a high density on the mesh, resulting in a point cloud.

(iii) Create a 2D rasterized mesh of the point cloud, where the raster cell height is the maximum elevation of all points in the cell.

(iv) Extrude the mesh surface in the negative Z direction to create a 3D volume (Fig. 4).

This procedure results in a watertight mesh which can be imported into FLOW-3D, with the limitation that if there are any overhangs in the model, they are converted to vertical faces due to the vertical rasterization step. This is a major limitation which will have implications for flow dynamics as erosion progresses over many time steps. Developing alternative robust solutions which allow overhangs is the subject of future work.

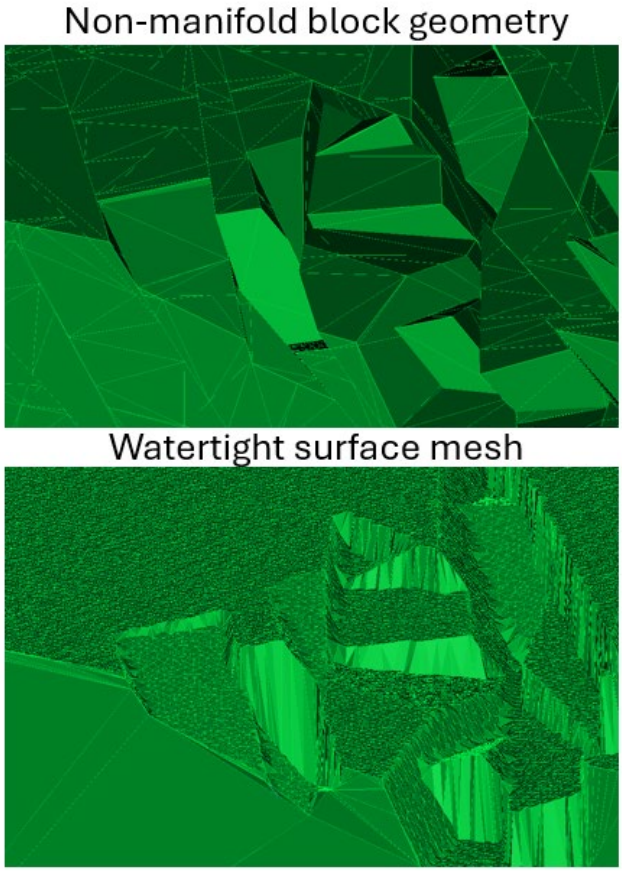

Fig. 4. Illustration of the process of merging block geometries into a single exterior surface for use in FLOW-3D.

2.3. Flow Velocity in FLOW-3D

FLOW-3D is a commercial CFD software that is commonly used in studies concerning water flow and erosion around dams, spillways, and other civil infrastructure (Demeke et al., 2019; Flow Science Inc., 2023; Hu et al., 2018). The FLOW-3D core model is based on a solver for the Navier-Stokes equations governing fluid flow.

In this study, we used FLOW-3D HYDRO version 2023R2 to compute flow velocities over the spillway surface and more specifically at removable block locations. The surface is imported and discretized using a rectilinear grid at a relatively coarse resolution of 2 meters. This resolution was chosen given the overall size of the model, the typical sizes of individual blocks, and the desired number of iterations to run to develop a proof of concept. For each surface needing to be analyzed, the simulation is advanced until flow velocities reach a steady state, after which the 3D velocity vectors above the surface are extracted, along with depth averaged velocity values at each grid cell. Each block is then assigned a velocity based on the grid cell closest to the block centroid.

Fluctuations in turbulence across a spillway surface and within the joints of the blocks are expected to manifest significant gradients in pressure and velocity in both the micro and macro structure of the flow. Accurately capturing these would lead to increased confidence in the prediction of mobility of the blocks as instantaneous peak estimates of resulting force would be captured. However, to quantitatively determine these fluctuations using CFD would require a computational mesh, corresponding time-step, and computing on a scale that would be prohibitively large for real-world applications. Fortunately, experimental results on polyhedral blocks with multiple degrees of freedom of motion suggest that using the average velocity predicted by industry standard turbulence closure models is adequate, even in relatively high turbulence conditions (George and Sitar, 2016). As such, we used steady-state solutions based on the RNG k-epsilon turbulence model to predict block mobility.

To simulate a flood hydrograph, multiple steady-state models would be used iteratively corresponding to different discharges at different time steps within a hydrograph.

2.4. Block Removability and Stability

For each block, given the geometry of its faces, which faces are exposed and not exposed, and the flow velocity in its vicinity, we created a Python function to compute whether the block is removable and unstable using analytical block theory concepts. Blocks created by *UnBlocks-gen* are convex, so we assume that all blocks are convex and apply block theory cases applicable to convex shapes. Otherwise, there is no limitation on the number of faces a single block can have. Terminology for a removable block is illustrated in Fig. 5.

According to Goodman and Shi (1985), a block is removable if the intersection of the joint (non-surface) plane halfspaces (the halfspace is the 3D space on the side of the joint plane facing towards the block interior) is not empty, or in other words, the joint faces are not tapered. Goodman and Shi demonstrated this is equivalent to stating that the system of joint halfspace linear inequalities Eq. (1) has a non-zero solution:

$$\vec{\boldsymbol{v}} \cdot \begin{bmatrix} x \\ y \\ z \end{bmatrix} \geq 0 \qquad (1)$$

Where $\boldsymbol{v}$ represents the *n x 3* matrix of blockside normal vectors for each of *n* joint planes in the block. The existence of a non-zero solution for each block is evaluated using linear programming.

For removable blocks, we compute block stability following the analytical method used by George and Sitar (2016).

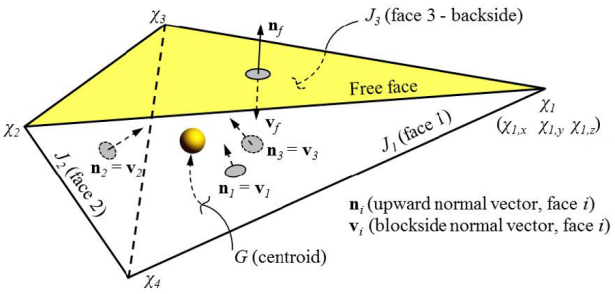

Fig. 5. Normal vectors and faces for a tetrahedral block (modified from George and Sitar (2016)).

First, using the nearby velocity extracted from FLOW-3D, the resultant force on the block is computed as follows (Eqn. (2)):

$$\mathbf{r} = \sum_i^n \frac{1}{2} \rho_w u^2 C_{pi} A_i \mathbf{v}_i + \mathbf{W_b} \qquad (2)$$

Where $\mathbf{r}$ is the resultant force vector, ρ is water density, u is the flow velocity magnitude in the vicinity of the block, A is the area of face i, $\boldsymbol{v}$ is the blockside normal vector, $\boldsymbol{W_b}$ is the buoyant weight of the block. C_{pi} represents the average dynamic pressure coefficient for face *i*. We compute the resultant by splitting this into three equations, one for each of the x, y, and z components.

The dynamic pressure coefficient C_p for each face enables estimating the internal forces acting to destabilize the block and is therefore critical to realizing realistic simulations of block erosion. The coefficients are known to be influenced by several factors, including the water flow direction above the block surface, the degree of protrusion of the block into flow, and the local surface and joint geometries around the block of interest (George & Sitar, 2016; Jalili Kashtiban et al., 2021; Reinius, 1986). For the initial demonstration of the model, we assign generic pressure coefficients of 0.4 to all joint faces and 0.05 to all surface faces when computing stability. Developing automated methods for estimating realistic pressure coefficients is the subject of future work. However, we note that in principle, all information necessary to do this, such as water velocity and block geometry, is contained within the existing block data structure.

Once the resultant force vector $\mathbf{r}$ is known, block stability can be evaluated for each of the three possible failure modes: lifting, one-plane sliding, and two-plane sliding. For each block, the failure mode is first evaluated, then the failure criterion for that specific failure mode is computed (see equations in George and Sitar, 2016, section 5.2.3). Note that for this demonstration we neglect rotational failure modes.

For lifting, the resultant force vector is pointing outward with respect to all joint planes ($\boldsymbol{r} \cdot \boldsymbol{v_i} > 0$). For one-plane sliding, $\mathbf{r}$ is pointing inward towards one joint plane and outward for all other joint planes. For two-plane sliding,

r is pointing inward towards both sliding planes and away from the remaining planes. For each failure mode, all possible combinations of faces are tested as potential sliding planes, and under normal circumstances (no overhangs) only one failure mode criteria for one combination of faces will be true.

Once the failure mode is determined, the stabilizing force F is calculated for that mode (George and Sitar, 2016).

For lifting:

$$F = |\mathbf{r}| \tag{3}$$

For one-plane sliding on plane i:

$$F_i = |\mathbf{n}_i \times \mathbf{r}| - |\mathbf{n}_i \cdot \mathbf{r}| \cdot \tan(\phi_i) \tag{4}$$

For two-plane sliding on planes i and j:

$$F_{ij} = \frac{1}{|\mathbf{n}_i \times \mathbf{n}_j|^2} \cdot \begin{bmatrix} |\mathbf{r} \cdot (\mathbf{n}_i \times \mathbf{n}_j)| \cdot |\mathbf{n}_i \times \mathbf{n}_j| - \\ |(\mathbf{r} \times \mathbf{n}_j) \cdot (\mathbf{n}_i \times \mathbf{n}_j)| \cdot \tan(\phi_i) - \\ |(\mathbf{r} \times \mathbf{n}_i) \cdot (\mathbf{n}_i \times \mathbf{n}_j)| \cdot \tan(\phi_j) \end{bmatrix} \tag{5}$$

If F is positive, the block is unstable and is removed in the next iteration of the surface. In the current version of the model, all joint surfaces are assigned the same friction angle (Φ), and no cohesion is included. In future versions, different friction angles could be specified for different joint sets.

3. RESULTS

We performed several tests using the proposed method to illustrate its ability to generate an eroded spillway surface. The first test involved repeatedly running the block stability and removal steps with fixed water velocities (i.e., not updating the velocity each step) to illustrate the model's ability to automatically identify new removable blocks. After initialization of an un-eroded surface, the block stability function evaluates which blocks are removable, and for removable blocks, computes their stability given current water velocity conditions. The surface update function then communicates which block faces are newly fully exposed to the surface given blocks that were just removed, and the process is then repeated.

Fig. 6 shows an example of four erosion steps, highlighting that as new blocks are exposed to the surface, some block orientations become removable while others remain non-removable. Fig. 7 illustrates the outputs of FLOW-3D on a hypothetical erosion event. In this example, surface geometry was created by removing blocks if they are adjacent to previously eroded blocks, starting from an initial set of blocks removed at the base of the model. Removable blocks tend to have a long axis trending northeast-southwest, showing the capability of the model for anisotropic erosion behavior that is controlled by the input discontinuity orientations.

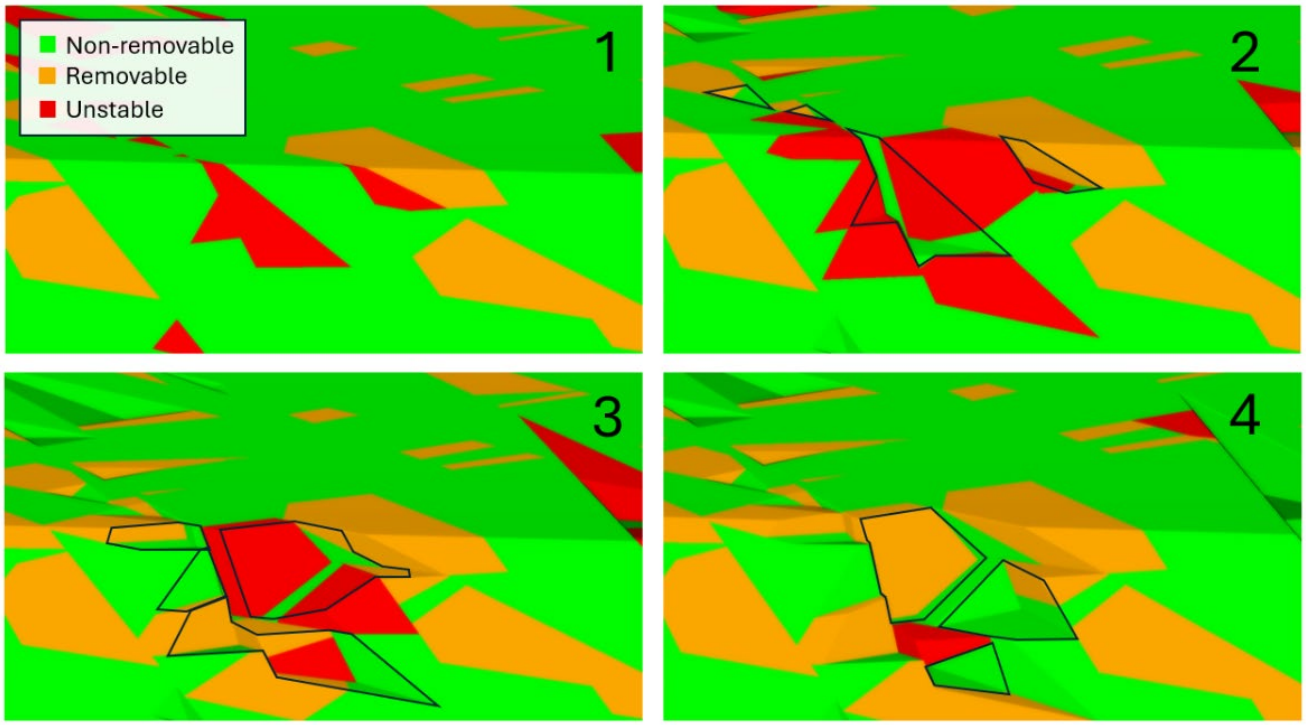

Fig. 6. Diagram showing the progression of block erosion through four time steps (1 through 4), as labeled in the upper right of each panel. Between panels, the removability and stability of each block is recalculated as unstable blocks from the previous step are removed (represented by black outlines). In this example a single static water velocity of 11 m/s is used for demonstration purposes. Note that unstable blocks are also necessarily removable.

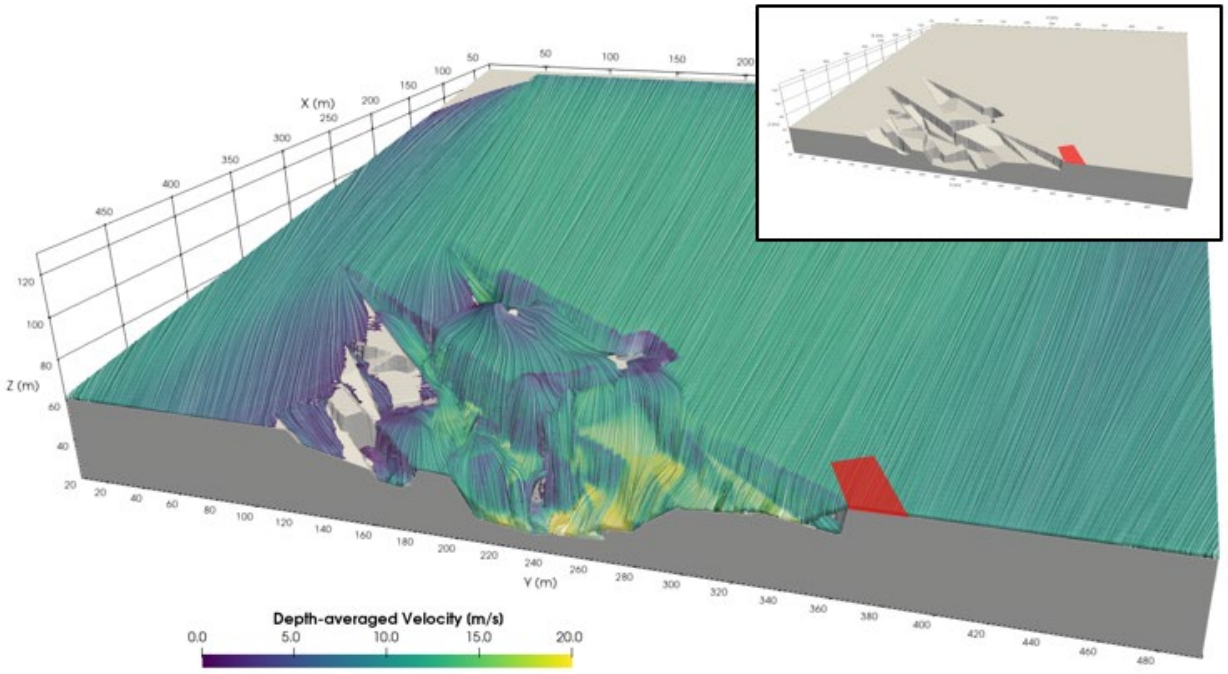

Fig. 7. Illustration of FLOW-3D outputs, showing depth-averaged velocity contours and flow lines. Underlying model geometry is shown on the upper right. Red polygon indicates block analyzed in Fig. 8.

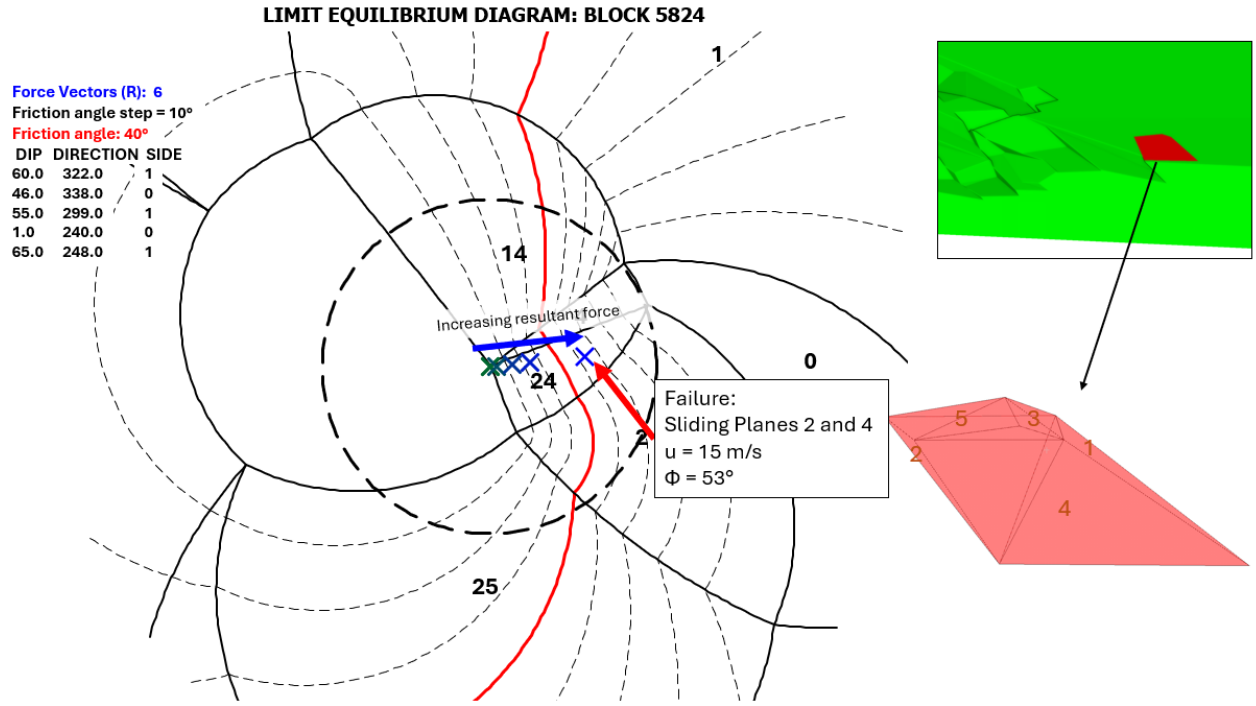

Fig. 8. Limit equilibrium stereonet plot after Shi (1992) showing resultant force vectors (blue X markers) for flow velocities from 0 to 15 m/s. Resultant forces plot in the region indicating a two-plane sliding failure mode alone planes 2 and 4. Wire mesh diagram to the right shows the location and geometry of the analyzed block and its face indices.

Detailed analyses of individual blocks can also be performed using this framework to better understand the failure modes. Fig. 8 illustrates a limit equilibrium stereonet (after Shi 1992) for a single removable block near the bottom of the model. Water velocity is increased until the block fails (a friction angle of 53 degrees and rock density of 2300 kg/m^3 are assumed). In this case, the block fails at a water velocity of approximately 15 m/s, with two-plane sliding on joint faces 2 and 4.

Fig. 9 illustrates an example of flow interactions with a cavity remaining from a removed block. It is evident that

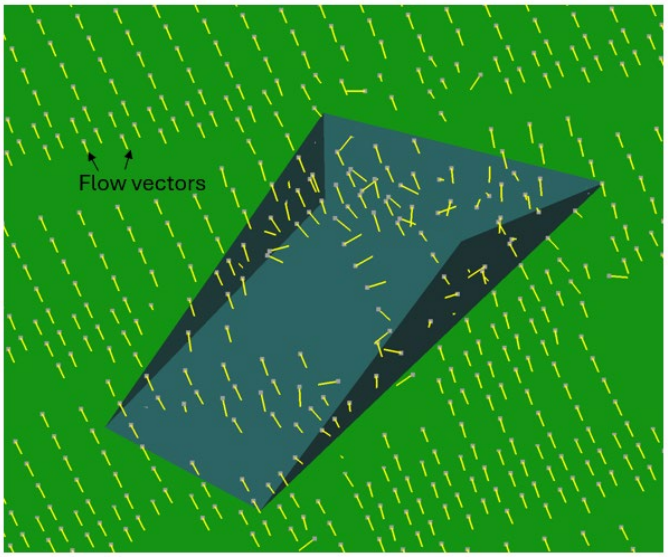

Fig. 9. Illustration of flow velocity vectors output from FLOW-3D simulation for a single removed block on a flat plane.

flow direction deviates significantly from the parallel flow outside of the block cavity, and there is also notable heterogeneity across the cavity due to the asymmetric depth of the block.

4. DISCUSSION AND CONCLUSIONS

This paper presents an approach which enables more realistic spillway-scale simulations of rock erosion. We used *UnBlocks-gen* to create a stochastic rock mass comprised of convex polyhedrons, then created a Python software package which computes the removability and stability of each block based on flow information computed using FLOW-3D.

The demonstrated model framework has several key benefits. The ability to compute stability for arbitrarily shaped blocks allows for complex representations of in-situ joint sets, 3D flow conditions, and resulting anisotropy in erosion across the slope. For example, in the case presented in Fig. 7, erosion progressed more quickly in the NE-SW direction compared to other directions – a behavior which was not pre-programmed.

The framework also has the potential to be more efficient than alternative methods for simulating thousands of blocks simultaneously. As noted by Gardner (2023), modeling the governing fluid dynamics equations on polyhedrons is not practical at the project scale due to their computational requirements. The computation of limit equilibrium stability equations is trivial in comparison, and flow velocities can be computed for the entire spillway surface using FLOW-3D. However, FLOW-3D's computational requirements are not insignificant either and depend on a variety of factors. We expect there will be tradeoffs between FLOW-3D model resolution, number of erosion steps, and simulation run time.

While the initial model results are promising, there are still several major improvements which need to be made before the framework can be used operationally. The simplification used to generate an exterior surface disallows the formation of overhangs, which limits the development of more complex flow behavior in plunge pools and knickpoints. Future model versions should allow for a more accurate surface to be passed to FLOW-3D.

This study also assumes constant joint and surface pressure coefficients across all blocks regardless of geometry, whereas it is known that joint pressure varies considerably depending on block protrusion and flow regime. In the future we intend to use the local geometry around a block to automatically infer the most likely pressure coefficient scenario to apply, but this will also require extensive review of previous work and further experiments with numerical simulations for calibration.

REFERENCES

1. Annandale, G. W. (1995). Erodibility. *Journal of Hydraulic Research*, *33*(4), 471–494. https://doi.org/10.1080/00221689509498656
2. Bollaert, E. F. R. (2012). Wall jet rock scour in plunge pools: A quasi-3D prediction model. *Hydropower & Dams*, *XX*.
3. Bollaert, E. F. R., & Schleiss, A. (Eds.). (2002). *Transient water pressures in joints and formation of rock scour due to high-velocity jet impact*. EPFL-LCH.
4. Bollaert, E. F. R., & Schleiss, A. J. (2005). Physically Based Model for Evaluation of Rock Scour due to High-Velocity Jet Impact. *Journal of Hydraulic Engineering*, *131*(3), 153–165. https://doi.org/10.1061/(ASCE)0733-9429(2005)131:3(153)
5. Boon, C. W., Houlsby, G. T., & Utili, S. (2015). A new rock slicing method based on linear programming. *Computers and Geotechnics*, *65*, 12–29. https://doi.org/10.1016/j.compgeo.2014.11.007
6. Demeke, G. K., Asfaw, D. H., & Shiferaw, Y. S. (2019). 3D Hydrodynamic Modelling Enhances the Design of Tendaho Dam Spillway, Ethiopia. *Water*, *11*(1), 82. https://doi.org/10.3390/w11010082
7. Flow Science Inc. (2023). *FLOW-3D HYDRO* (Version 2023R2) [Computer software]. https://www.flow3d.com
8. Gardner, M. (2023). Toward a Complete Kinematic Description of Hydraulic Plucking of Fractured Rock. *Journal of Hydraulic Engineering*, *149*(7), 04023015. https://doi.org/10.1061/JHEND8.HYENG-13193
9. Gardner, M., Kolb, J., & Sitar, N. (2017). Parallel and scalable block system generation. *Computers and Geotechnics*, *89*, 168–178. https://doi.org/10.1016/j.compgeo.2017.05.001

10. Gardner, M., & Sitar, N. (2019). Modeling of Dynamic Rock–Fluid Interaction Using Coupled 3-D Discrete Element and Lattice Boltzmann Methods. *Rock Mechanics and Rock Engineering*, *52*(12), 5161–5180. https://doi.org/10.1007/s00603-019-01857-x
11. George, M. (2023, October). *A Block Theory approach for rock erodibility assessment incorporating 3D high-resolution site characterization data*. 15th ISRM Congress 2023 & 72nd Geomechanics Colloquium., Salzburg, Austria.
12. George, M., & Sitar, N. (2016). *3D Block Erodibility: Dynamics of Rock-Water Interaction in Rock Scour.* https://doi.org/10.13140/RG.2.2.29909.52965
13. Goodman, R. E., & Shi, G. (1985). *Block theory and its application to rock engineering*. Prentice-Hall Englewood Cliffs, N.J.; WorldCat.
14. Hu, H., Zhang, J., & Li, T. (2018). Dam-Break Flows: Comparison between Flow-3D, MIKE 3 FM, and Analytical Solutions with Experimental Data. *Applied Sciences*, *8*(12), 2456. https://doi.org/10.3390/app8122456
15. Hyman, J. D., Karra, S., Makedonska, N., Gable, C. W., Painter, S. L., & Viswanathan, H. S. (2015). dfnWorks: A discrete fracture network framework for modeling subsurface flow and transport. *Computers & Geosciences*, *84*, 10–19. https://doi.org/10.1016/j.cageo.2015.08.001
16. Ivars, D., Pierce, M. E., Darcel, C., Reyes-Montes, J., Potyondy, D. O., Paul Young, R., & Cundall, P. A. (2011). The synthetic rock mass approach for jointed rock mass modelling. *International Journal of Rock Mechanics and Mining Sciences*, *48*(2), 219–244. https://doi.org/10.1016/j.ijrmms.2010.11.014
17. Jalili Kashtiban, Y., Saeidi, A., Farinas, M.-I., & Quirion, M. (2021). A Review on Existing Methods to Assess Hydraulic Erodibility Downstream of Dam Spillways. *Water*, *13*(22), Article 22. https://doi.org/10.3390/w13223205
18. Rasmussen, L. L. (2020). U n B l o c k s g e n: A Python library for 3D rock mass generation and analysis. *SoftwareX*, *12*. https://doi.org/10.1016/j.softx.2020.100577
19. Reinius, E. (1986). Rock Erosion. In *Water Power and Dam Construction* (pp. 43–48).
20. Shi, G.-H. (1992). Discontinuous Deformation Analysis: A New Numerical Model For The Statics And Dynamics Of Deformable Block Structures. *Engineering Computations*, *9*(2), 157–168. https://doi.org/10.1108/eb023855
21. Teng, P., Johansson, F., & Hellström, J. G. I. (2023). Modelling erosion of a single rock block using a coupled CFD-DEM approach. *Journal of Rock Mechanics and Geotechnical Engineering*, *15*(9), 2375–2387. https://doi.org/10.1016/j.jrmge.2023.06.001