

박사학위논문

Ph.D. Dissertation

토석류 사방구조물 최적 위치선정 및  
성능평가 모델

A New Simulation Model for Optimal Location Selection and  
Performance Evaluation of Barriers as  
Mitigation Against Debris Flow

2022

전예녹 (全예녹 Cheon, Enok)

한국과학기술원

Korea Advanced Institute of Science and Technology

박사학위논문

토석류 사방구조물 최적 위치선정 및  
성능평가 모델

2022

전에녹

한국과학기술원

건설 및 환경공학과

# A New Simulation Model for Optimal Location Selection and Performance Evaluation of Barriers as Mitigation Against Debris Flow

전에녹

위 논문은 한국과학기술원 박사학위논문으로  
학위논문 심사위원회의 심사를 통과하였음

2022년 6월 14일

심사위원장 이승래



심사위원 조계춘



심사위원 권태혁



심사위원 주진현



심사위원 김윤태



# A New Simulation Model for Optimal Location Selection and Performance Evaluation of Barriers as Mitigation Against Debris Flow

Enok Cheon

Advisor: Seung-Rae Lee

A dissertation/thesis submitted to the faculty of  
Korea Advanced Institute of Science and Technology in  
partial fulfillment of the requirements for the degree of  
Doctor of Philosophy in Civil and Environmental Engineering

Daejeon, Korea  
June 14, 2022

Approved by



Seung-Rae Lee

Professor of Civil and Environmental Engineering

The study was conducted in accordance with Code of Research Ethics<sup>1)</sup>.

---

1) Declaration of Ethical Conduct in Research: I, as a graduate student of Korea Advanced Institute of Science and Technology, hereby declare that I have not committed any act that may damage the credibility of my research. This includes, but is not limited to, falsification, thesis written by someone else, distortion of research findings, and plagiarism. I confirm that my dissertation contains honest conclusions based on my own careful research under the guidance of my advisor.

## DCE

전에녹. 토석류 사방구조물 최적 위치선정 및 성능평가 모델. 건설 및 환경공학과. 2022년. 127+viii 쪽.

지도교수: 이승래. (영문 논문)

Enok Cheon. A New Simulation Model for Optimal Location Selection and Performance Evaluation of Barriers as Mitigation Against Debris Flow. Department of Civil and Environmental Engineering. 2022. 127+viii pages. Advisor: Seung-Rae Lee. (Text in English)

### 초록

사방구조물을 통해 토석류 피해를 저감하기 위해서는 효율적인 최적 사방구조물 설계 방법과 사방구조물 성능평가 수치해석 방법이 필수적이다. 본 연구에서는 새롭게 개발된 SPEC-debris-barrier 플랫폼은 토석류 거동특성 분석, 최적 사방구조물 설계와 폐쇄형 및 열린형 사방구조물 성능평가 해석을 실행할 수 있는 다목적 해석 소프트웨어이다. 개발된 플랫폼은 다양한 폐쇄형 및 열린형 사방구조물을 모델링할 수 있으며 CPU 멀티프로세싱을 활용해 연산 속도를 높이는 기능이 있다. 또한, 플랫폼은 2D 및 3D 웹 기반 대화형 표와 애니메이션을 생성하여 해석 결과를 시각화 한다. SPEC-debris 시뮬레이션은 SPEC-debris-barrier 플랫폼에 내장된 토석류 수치해석 방법이다. 이 방법은 "등가 유체" 원리를 따르는 메시리스 수치해석 방법으로. (a) smoothed particle hydrodynamic 기법, (b) 얇은 물 방정식에서 유래된 경로 알고리듬, (c) 에너지 보존 법칙 모델, (d) 충돌 메커니즘 등을 적용한다. SPEC-debris 시뮬레이션은 선형 운동량의 보전 법칙 및 반발 계수를 활용하여 사방구조물과의 충돌 후 운동 에너지의 손실 해석이 가능하다. SPEC-debris-barrier 플랫폼에 통합된 다른 설계 도구는 최적 사방구조물 설계 프레임워크이다. 이 프레임워크는 클러스터 분석을 통해 SPEC-debris 수치해석 입자들을 클러스터로 분류한 다음, 토석류 클러스터 데이터를 토석류 네트워크로 구조화한다. 네트워크 데이터 구조 및 확률적 타부 탐색 최적화 알고리즘을 활용해 여러 최적화 기준을 충족하는 최적의 폐쇄형 및 열린형 사방구조물 위치를 결정한다. 토석류 거동특성 평가 및 장벽 설계를 위한 SPEC-debris-barrier 플랫폼의 성능 평가와 적용 가능성은 소형 플룸 실험과 2011년 서울 우면산 토석류 사례를 역분석하여 검증하였다. 또한, 민감도 분석으로 최적의 폐쇄형 및 열린형 사방구조물 위치 선정의 사용성을 입증했다. 지속적인 후방 분석과 정확한 물성치 예측이 가능한 모델들의 개발을 통해 SPEC-debris-barrier 플랫폼은 토석류 시뮬레이션 및 사방구조물 설계를 위하여 최상의 소프트웨어 제품이 될 것이라 예상된다.

핵심 낱말 토석류; 최적화; 사방구조물 최적위치; 사방구조물 성능평가; 충돌; 소프트웨어

## **Abstract**

To mitigate the risk of debris-flow hazards through barriers, efficient methods for designing optimal barriers and debris-flow analysis methods that can evaluate barrier performance are essential. The newly-developed SPEC-debris-barrier platform is multi-purpose analysis software capable of analyzing debris-flow propagation, designing optimal barriers, and evaluating closed and open-type barrier performance. The developed platform can model various barriers and utilize CPU multiprocessing to enhance computational speed. Furthermore, the platform generates 2D and 3D web-based interactive plots and animations to visualize the analysis results. The SPEC-debris simulation is a numerical analysis method incorporated into the SPEC-debris-barrier platform. The method is a meshless numerical method that follows the “equivalent fluid” principle. The following equations govern the SPEC-debris model: (a) smoothing particle hydrodynamic interpolation, (b) pathway algorithm adopted from shallow-water equations, (c) energy conservation model, and (d) collision mechanism. The SPEC-debris simulation utilizes coefficient of restitution and conservation of linear momentum to simulate the loss of kinetic energy after collision with a barrier. The optimal barrier design framework incorporated in the SPEC-debris-barrier platform is used to determine optimal barrier locations. The framework compiles the particles from the SPEC-debris method into clusters through cluster analysis, then stores the debris-flow cluster data into a network data structure. The network data structure and stochastic tabu search heuristic optimization algorithm search optimal closed and open-type barrier locations that satisfy multiple optimization criteria. The capability and applicability of the SPEC-debris-barrier platform for assessing debris-flow and designing barriers were tested and verified through back-analyses of flume test and the 2011 debris-flow incident at Mt. Umyeon, Seoul. In addition, the sensitivity analysis demonstrated the application of selecting optimal closed and open-type barrier locations. With continuous back-analyses and accurate prediction of material properties, the SPEC-debris-barrier platform should be the ideal software suite for runout simulation and mitigation design.

**Keywords** Debris-flow; Optimization; Optimal Barrier Location; Barrier Performance; Collision; Software

# Contents

Contents .....	i
List of Tables .....	v
List of Figures .....	vi
<b>Chapter 1. Introduction</b>	<b>1</b>
1.1 Problem statement .....	1
1.2 Research objectives .....	3
1.3 Scope-organization .....	4
<b>Chapter 2. SPEC-debris Analysis Method</b>	<b>5</b>
2.1 Introduction .....	5
2.2 Governing Equations .....	6
2.2.1 Overview .....	6
2.2.2 Smoothed particle hydrodynamics (SPH) .....	8
2.2.2.1 Theory .....	8
2.2.2.2 Depth and depth gradient interpolation .....	9
2.2.2.3 Adaptive smoothing length .....	11
2.2.2.4 Neighboring particles .....	11
2.2.2.5 Boundary ghost particles .....	12
2.2.3 Pathway model .....	13
2.2.3.1 Gravity .....	13
2.2.3.2 Hydrodynamic pressure gradient .....	15
2.2.3.3 Basal rheological resistance .....	19
2.2.3.4 Entrainment momentum transfer .....	19
2.2.3.5 Pathway algorithm .....	20
2.2.4 Energy conservation .....	22
2.2.4.1 Entrainment model .....	22
2.2.4.2 Kinetic energy .....	23
2.2.4.3 Gravitational potential energy .....	23
2.2.4.4 Pressure potential energy .....	24
2.2.4.5 Rheological frictional energy loss .....	24
2.2.4.6 Energy conservation model .....	25
2.2.4.7 Particle displacement without wall collision .....	26
2.2.4.8 Particle displacement along the wall .....	28

2.2.5 Collision .....	28
2.2.5.1 Particle-wall collision .....	28
2.2.5.2 Particle-particle collision .....	29
2.4 Summary .....	30
<b>Chapter 3. Optimal Barrier Design Framework</b>	<b>31</b>
3.1 Introduction .....	31
3.2 Review of Existing Barrier Design Framework .....	32
3.2.1 Empirical analysis approach .....	32
3.2.2 Numerical analysis approach .....	33
3.3 Optimal Barrier Design Framework .....	34
3.3.1 Framework overview .....	34
3.3.2 Debris-flow 3D simulation .....	35
3.3.3 Debris-flow cluster analysis .....	36
3.3.3.1 Clustering debris-flow particles .....	36
3.3.3.2 Debris-flow cluster boundary .....	39
3.3.3.3 Debris-flow cluster representative quantities .....	41
3.3.3.4 Debris-flow cluster merge and separation .....	43
3.3.3.5 Outputs of debris-flow cluster analysis .....	44
3.3.4 Debris-flow network .....	45
3.3.4.1 Network structure .....	45
3.3.4.2 Node type .....	46
3.3.4.3 Closed-type barrier mitigation check .....	47
3.3.4.4 Opened-type barrier mitigation check .....	48
3.3.5 Optimal barrier location selection .....	50
3.3.5.1 Optimization procedure .....	51
3.3.5.2 Optimization criteria and cost analysis .....	52
3.3.5.3 Exclusion of barrier location candidates .....	54
3.4 Summary .....	54
<b>Chapter 4. SPEC-debris-barrier Platform</b>	<b>55</b>
4.1 Introduction .....	55
4.2 Platform Features .....	55
4.2.1 Analysis overview .....	55
4.2.2 User interface .....	55
4.2.3 Instruction .....	58

4.3 Input .....	61
4.3.1 JSON file format .....	61
4.4 Output .....	62
4.4.1 Analysis results: JSON .....	62
4.4.2 Interactive HTML plots .....	62
4.4.2.1 Scatter, heatmap, and animation in 2D and 3D .....	62
4.4.2.2 Interactive features .....	65
4.5 Discussion and Conclusion .....	65
<b>Chapter 5. Tests and Case Studies</b>	<b>67</b>
5.1 Introduction .....	67
5.2 Barrier performance .....	67
5.2.1 Description .....	67
5.2.2 Methodology .....	68
5.2.3 Results and discussion: barrier performance .....	69
5.2.4 Results and discussion: particle number per cell .....	71
5.2.5 Results and discussion: barrier effect .....	72
5.3 Case Study: Mt. Umeyeon .....	73
5.3.1 Description .....	73
5.3.2 Methodology .....	73
5.3.3 Results and discussion: debris-flow analysis .....	74
5.3.4 Results and discussion: optimal closed-type barrier .....	77
5.3.5 Results and discussion: optimal opened-type barrier .....	79
5.4 Conclusion .....	83
<b>Chapter 6. Conclusions and Recommendations</b>	<b>84</b>
6.1 Conclusion and summary .....	84
6.2 Recommendation for further studies .....	86
<b>Appendix A. SPEC-debris-barrier Platform JSON</b>	<b>88</b>
A.1 Input JSON .....	88
A.2 Basic information .....	89
A.3 DEM, road, and structures .....	90
A.3.1 Topography, source, and material .....	90
A.3.2 Road .....	93
A.3.3 Structures .....	93
A.4 Initial and material properties .....	97
A.4.1 Initial velocity .....	98

A.4.2 Rheology, internal friction, and entrainment .....	98
A.4.3 Coefficient of restitution (COR) .....	100
A.4.4 Initial SPH interpolation .....	100
A.5 Algorithm options .....	101
A.5.1 Interpolation method .....	101
A.5.2 Entrainment model .....	101
A.5.3 Cluster boundary algorithm .....	102
A.6 Control parameters .....	102
A.6.1 Time .....	102
A.6.2 Initial particle .....	102
A.6.3 Gravity and free-fall .....	105
A.6.4 Steepest local gradient descent cell size .....	105
A.6.5 SPH smoothing length parameter .....	106
A.6.6 Cluster analysis parameter .....	106
A.6.7 Multiprocessing parameter .....	108
A.7 Optimal barrier location selection parameters .....	108
A.7.1 Search parameters .....	108
A.7.2 Closed-type barrier optimization .....	110
A.7.3 Combined open and closed-type barrier optimization .....	110
A.8 Plot options .....	112
A.8.1 Static plot option .....	112
A.8.2 Animation option .....	113
A.8.3 Maximum legend .....	114
A.8.4 Plot and figure size .....	115
A.9 Output options .....	116
A.8.1 output file options .....	116
A.8.2 decimal points .....	117
A.10 SPEC-debris cluster .....	117
A.11 Output JSON .....	117
Bibliography .....	119
Acknowledgments .....	124
Curriculum Vitae .....	125
Research Achievements .....	126

## List of Tables

Table 2.1 Limitation of current 3D numerical analysis runout models

Table 4.1 Analysis keywords and their corresponding analyses

Table 5.1 P00 slit-type barrier model input

Table 5.2 Barrier performance results based on *particle number per cell*

Table 5.3 Material properties for Mt. Umyeon simulation

Table 5.4 Optimization criteria weighting factors

Table 5.5 Comparison of simulation and observed value for Mt. Umyeon

Table 5.6 Debris-flow cluster characteristics at each optimal closed-type barrier location

Table 5.7 Open-type barrier performance for sensitivity analysis

Table A.1 Input categories and valid variable names

Table A.2 JSON data type and syntax

## List of Figures

- Figure 1.1 Damage caused by landslides and debris-flows in Korea from 1976 to 2020
- Figure 1.2 Number of annual debris-flow barrier installations since 1986
- Figure 2.1 Diagram of fluid governed by the St. Venant equations
- Figure 2.2 Effect of barrier on selecting neighboring particles in SPH interpolation
- Figure 2.3 Fixed ghost particles outside the DEM boundary for the SPH interpolation
- Figure 2.4 Local particle terrain (yellow region) and local linear regression descent ( $\vec{\beta}$ ): (a) no presence of structures and (b) presence of structures
- Figure 2.5 Particle movement gradients ( $\vec{V}$ ) computed by the pathway algorithm
- Figure 2.6 General energy conservation model for runout analysis
- Figure 2.7 Energy loss from the variable frictional force
- Figure 2.8 Procedure for particle velocity and position at current time step
- Figure 2.9 Collision between fluid particles and wall
- Figure 2.10 Collision between two fluid particles
- Figure 3.1 Interdependent factors for designing optimal barriers
- Figure 3.2 Barrier design scenarios and DAN3D debris-flow simulation results (Choi and Kwon, 2021)
- Figure 3.3 Optimal barrier design framework flowchart
- Figure 3.4 Debris-flow cluster as a collection of debris-flow particles
- Figure 3.5 Overall flowchart of debris-flow cluster analysis (Cheon et al., 2020)
- Figure 3.6 A simple demonstration of the k-means clustering algorithm (Weston.pace, 2007): (a) initialization, (b) partition, (c) new centroid, and (d) termination.
- Figure 3.7 Comparison of cluster boundary from the Convex Hull (dotted line) and the  $\alpha$ -shape (solid line) algorithm (Cheon et al., 2020)
- Figure 3.8 Exclusion of outlier point (red circle) in cluster boundary through the selection of alpha value (Cheon et al., 2020)
- Figure 3.9 The definition of the width of a debris-flow cluster (Cheon et al., 2019)
- Figure 3.10 The definition of the travel distance (S) and distance-from-road (D)
- Figure 3.11 Criteria to determine the merging of two debris-flow clusters (Cheon et al., 2020)
- Figure 3.12 Criteria to determine the splitting of a debris-flow cluster (Cheon et al., 2020)
- Figure 3.13 Simplified debris-flow network diagram
- Figure 3.14 Debris-flow cluster location (DFL) with merge node type

- Figure 3.15 Debris-flow cluster location (DFL) with multi-impact node type
- Figure 3.16 Example of successful mitigation for closed-type barrier with debris-flow network data structure
- Figure 3.17 Example of mitigation check for open-type barrier with debris-flow network data structure
- Figure 3.18 Overall flowchart of optimal barrier location selection
- Figure 3.19 Local and global minimum solution
- Figure 3.20 Procedure of stochastic tabu search optimization algorithm for optimal barrier location selection
- Figure 4.1 Overall flowchart of the SPEC-debris-barrier platform integrating SPEC-debris method (yellow box), debris-flow cluster analysis (blue box), and optimal barrier location framework (green box)
- Figure 4.2 User interface of SPEC-debris-barrier platform
- Figure 4.3 Example of an error message pop-up window in the SPEC-debris-barrier platform
- Figure 4.4 Selecting the “Add Python 3.xx to PATH” option during Python3 installation
- Figure 4.5 Loading input file: (a) opening input JSON file, and (b) successful loading of an input JSON file
- Figure 4.6 Pop-up window after checking to inquire whether to proceed to the analyses
- Figure 4.7 Command prompt (cmd) analyses progress message
- Figure 4.8 Analyses completion message on: (a) pop-up window and (b) operation status
- Figure 4.9 Template of the input JSON file
- Figure 4.10 Topography plot in (a) 2D as contour maps and (b) 3D as a surface plot
- Figure 4.11 Debris-flow cluster static plot examples: (a) cluster flowpath in 2D, (b) cluster flowpath in 3D, (c) average cluster velocity in 2D, and (d) average cluster velocity in 3D
- Figure 4.12 Examples of maximum recorded debris-flow particle velocity heatmap plot with (a) only the particle results and (b) both particle and cluster results
- Figure 4.13 Example of debris-flow particle animation with (a) only the particle results and (c) both particle and cluster results
- Figure 4.14 Interactive plot features: (a) in-built feature buttons and (b) playback features for animations
- Figure 5.1 P00 slit-type barrier
- Figure 5.2 Small-scale flume experimental setup: (a) section view and (b) plan view (Choi et al., 2018)
- Figure 5.3 SPEC-debris simulation snapshots of reference case: (a)  $t = 0$ , (b)  $t = 0.2\text{s}$ , (c)  $t = 0.4\text{s}$ , and (d)  $t = 0.6\text{s}$
- Figure 5.4 SPEC-debris simulation snapshots of P00 slit-type wall: (a)  $t = 0$ , (b)  $t = 0.2\text{s}$ , (c)  $t = 0.4\text{s}$ , (d)  $t = 0.6\text{s}$ , and (e)  $t = 0.8\text{s}$
- Figure 5.5 Reduction of debris-flow front speed after the moment of collision (marked by blue dash line)
- Figure 5.6 Particle collision against P00 slit-type wall: (a)  $t = 0.38\text{s}$  and (b)  $t = 0.40\text{s}$
- Figure 5.7 Debris-flow that affected the Raemian and Sindonga apartment complex (Lee et al., 2019)
- Figure 5.8 Debris-flow simulation results for the Raemian region: (a) max velocity and (b) max depth
- Figure 5.9 Debris-flow simulation results for the Sindonga region: (a) max velocity and (b) max depth
- Figure 5.10 Optimal closed-type barrier locations (marked by the red cross) based on the optimization criteria

weighting factor Cases 1, 2, and 3 (road marked by the yellow line)

Figure 5.11 Optimal open and closed-type barrier locations based on open-type barrier performance and the optimization criteria: (a) case 4 – low performance, (b) case 4 – high performance, (c) case 5 – low performance, (d) case 5 – high performance, (e) case 6 – low performance, and (f) case 6 – high performance

Figure 5.12 Optimal open and closed-type barrier locations based on open-type barrier performance and the optimization criteria at Mt. Umeyeon Sindonga region: (a) low performance and (b) high performance

Figure A.1 Flowpath DEM elevation modification procedure: (a) structure boundary, (b) modification of elevation inside structure boundary, (c) Zin modification, (d)  $\Delta Z$  modification, (e) min  $\Delta Z$  modification, and (f) max  $\Delta Z$  modification

Figure A.2 Barrier and design parameters: (a) closed-type barriers (“P”), (b) box-shaped baffles (“P”), (c) parallel slit-type barriers (“P”), (d) V-shape arranged slit-type barriers (“V”), (e) cylinder baffles (“C”), and (f) general building (“BD”)

Figure A.3 Source data displaying cells with non-zero depth (NZD) values

Figure A.4 Initial particle placement: (a) identify NZD cells, (b) classify NZD cells into source regions, (c) subdivide NZD cells and place particles

# Chapter 1. Introduction

## 1.1 Problem statement

Debris-flow is a rapidly-moving fluid-like slurry composed of water, landslide failure mass, and other eroded materials transported by the debris. It is considered one of the most devastating geohazards, resulting in causalities and infrastructure damage (Jakob and Hungr, 2005). The geography of Korea is composed of approximately 63% mountainous regions covered with soil layers, of which thickness usually does not exceed 4 m, with underlying bedrock (Park et al., 2016; Lee et al., 2019). In addition, more than half of the annual rainfall in the central region of Korea, which is approximately 1200-1500 mm, falls during the monsoon season from July to September. The shallow soil layer, mountainous terrain, and extreme rainfall during the monsoon season often cause flow-like landslide hazards (Vasu et al., 2016, Lee et al., 2020). With climate changes increasing the intensity and the frequency of extreme rainstorms (Crozier, 2010), the frequency of the massive debris-flow hazards has been increasing since the year 2000, as shown in Figure 1.1. In addition, since the 2000s, the average annual recovery cost spent for restoration from landslide damage was approximately 67 billion KRW.

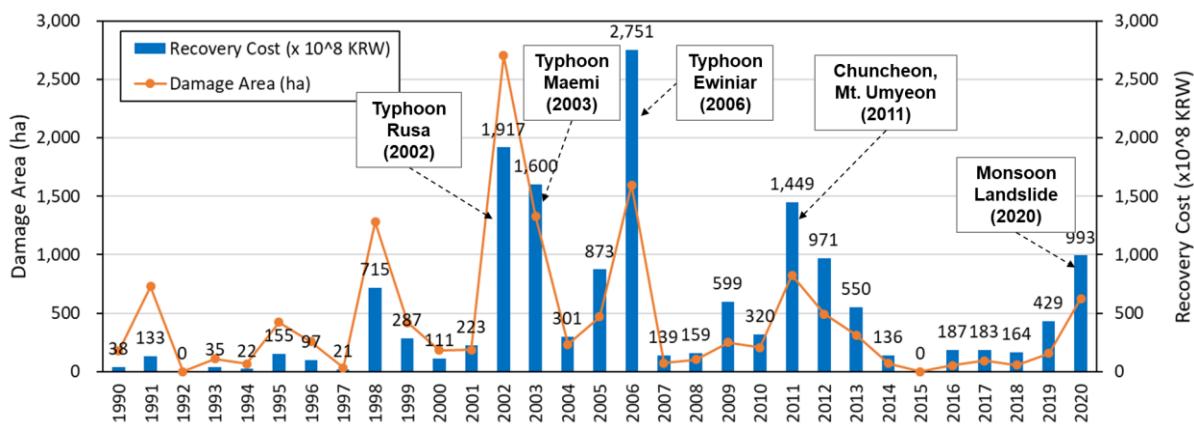


Figure 1.1 Damage caused by landslides and debris-flows in Korea from 1976 to 2020

(Data acquired from Korea Forest Service)

Since 2006, the Korean government has emphasized improving protection against landslide and debris-flow hazards to prevent damage from massive landslide and debris-flow disasters. Therefore, facilities, such as barriers, have been constructed to prevent similar disasters in densely populated cities and near critical infrastructures. As shown in Figure 1.2, an average of over 850 facilities were installed nationwide per year to restore and mitigate landslide and debris-flow hazards. In addition, the Korean Forest Service announced plans to expand the number of barriers to 24,600 by 2030.

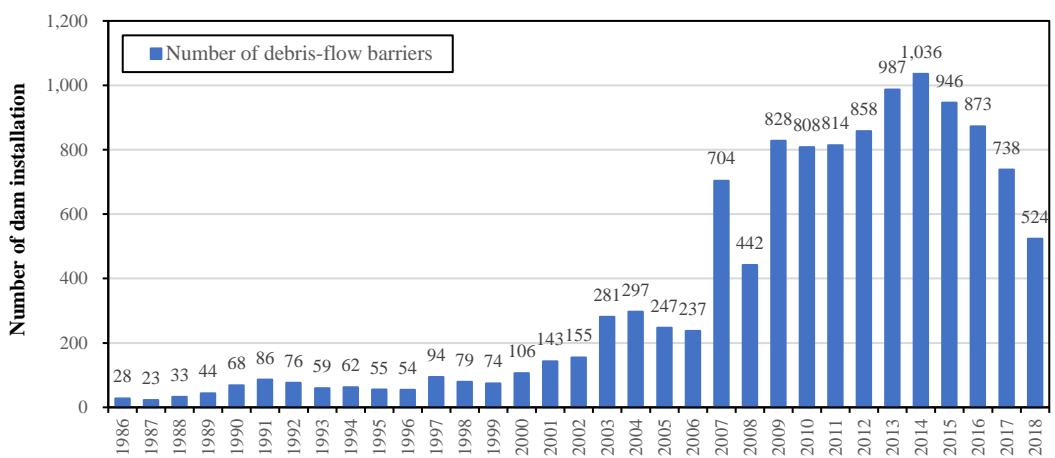


Figure 1.2 Number of annual debris-flow barrier installations since 1986

(Data acquired from Korea Forest Service)

Despite such efforts, domestic debris-flow mitigation method through the installation of open and closed-type barriers has not developed to the point that the key technology can be used to design sustainable barrier systems. The barrier installation locations are usually based on simple guidelines and engineering judgment, and the barrier designs are seldom optimized to maximize the cost-benefit ratio. The conservative approach to barrier design may lead to over-designed barriers that negatively impact the environment and produce an unappealing mountain landscape. Furthermore, the cost of constructing and maintaining the over-designed barriers could be financially unsustainable, resulting in unmaintained and ineffective barriers to mitigate debris-flow disasters. Thus, it is essential to develop core technologies that can efficiently design optimal barriers, i.e., most sustainable barriers, as mitigation against debris flow hazards. Governments and engineers can utilize these core technologies for the cost-effective design of open and closed-type barriers in potential debris-flow risk zones.

## **1.2 Research objectives**

This study aims to develop a computational tool for simulating debris-flow runout and designing optimal open and closed-type barriers to mitigate debris-flow disasters. The developed tool contains a novel debris-flow simulation method that will help understand how the debris-flows behave and the effect of barriers on debris-flow propagation. Additionally, the optimal barrier design framework in the developed tools can suggest a suitable and effective location for barriers for debris-flow prevention. Ultimately, this study aims to provide an efficient debris-flow disaster-mitigation platform for a disaster-safe society with an effective mitigation system against debris-flow. In order to achieve the main objective, several sub-objectives are suggested as follows:

1. Development of a runout model for analyzing debris-flow and computing barrier performance
  - Develop suitable assumptions and models for analyzing debris-flow runout simulation
  - Develop governing equations for computing propagation of debris-flow
  - Develop a suitable method for computing the depth of debris-flow
  - Develop an appropriate collision detection method
  - Develop the governing equations simulating the energy loss of debris-flow after collision with barriers
2. Development of a framework for designing optimal open and closed-type barriers
  - Develop a suitable analysis method for condensing the debris-flow simulation into essential data
  - Select a suitable data structure for storing the debris-flow simulation
  - Develop a quick checking method for determining the mitigation successfulness of the barriers
  - Develop optimization procedure for design barrier based on multiple design criteria
3. Development of the computational software for optimal barrier design
  - Develop functions to read, save, and edit the terrain data
  - Develop a simple and intuitive graphical user interface (GUI)
  - Construct a suitable file format to use as the input for the developed software
  - Develop plots and animations to visualize the debris-flow simulation
4. Validation of the developed computational tool
  - Validate the developed runout model with case studies
  - Validate the barrier performance analysis with laboratory experiments
  - Perform sensitivity analysis of the input parameters

### **1.3 Scope-organization**

This research highlights the development of the SPEC-debris-barrier platform, a software capable of executing runout simulation and optimal barrier design framework to perform debris-flow hazard assessment, evaluate barrier performances, and design optimal closed and open-type barriers against debris-flow disasters. This thesis covers the background, theory, methodology, and evaluation of the SPEC-debris-barrier platform. The version of the platform is the latest as of June 2022. Although the specific programming details of the developed SPEC-debris-barrier are not described in detail, the thesis provides a concise explanation of the underlying assumptions, principles, and sequence of the SPEC-debris-barrier platform.

Chapter 1 of this manuscript provides an introduction that summarizes the present problems and research objectives. The final Chapter 6 presents the research summaries, conclusions, limitations, and suggestions for future study. The other four chapters are summarized as follows:

Chapter 2 describes the development of the SPEC-debris method, a runout simulation model. A literature review on the existing runout simulation methods is introduced, then describes how the developed SPEC-debris method overcomes some of the limitations of the existing methods. The assumptions and the methodology of the developed method are detailed for the rest of the chapter. The developed simulation method is incorporated into the SPEC-debris-barrier platform, as described in Chapter 4.

Chapter 3 details the framework for designing optimal open and closed-type barriers that satisfy multiple design factors. The design factors are selected to optimize the components of sustainability (economic, social, and environmental aspects) of the barrier design. The chapter describes the process of optimal barrier design framework. The results from the SPEC-debris simulation method are converted to a data structure that a computer can utilize it to select optimal barrier locations. The developed design framework is incorporated into the SPEC-debris-barrier platform, as described in Chapter 4.

Chapter 4 provides a user manual for the SPEC-debris-barrier platform. This chapter aims to assist engineers planning to use the developed platform for their research purposes. The chapter describes the main features, operation instructions, and guides for creating input files. Additionally, the chapter describes algorithms and programming utilized to perform specific functions not described in Chapters 2 and 3.

Lastly, Chapter 5 validates the SPEC-debris-barrier platform. The reliability and accuracy of the SPEC-debris described in Chapter 2 are validated through back-analysis of case studies and laboratory experiments. The feasibility of the developed optimal barrier design framework is demonstrated by performing a parametric study using a debris-flow case study.

## Chapter 2. SPEC-debris Analysis Method

### 2.1 Introduction

A landslide runout analysis has been the standard approach for simulating the propagation of flow-like geohazards, such as debris-flow and avalanche, for back-analyzing previous hazards and predicting the flow of potential hazards. The runout analysis results provide valuable forensic investigations, risk assessment, and mitigation design data.

Due to the importance of the runout analysis, various researchers have developed three-dimensional (3D) runout models (Aurgarde et al., 2021; McDougall, 2017) based on empirical methods (Iverson et al. 1998; Rickenmann, 1999; Marchi, L. and D'Agostino, 2004; Berti and Simoni, 2014), energy conservation spreading models (Horton et al., 2013), depth-averaged continuum models (Hung, 1995; McDougall, 2006; FLO-2D Software Inc., 2007; Hung and McDougall, 2009; Christen et al., 2010; Pastor et al., 2014; Mergili et al., 2017), continuum models (Soga et al., 2016; Jeong and Lee, 2019; Domínguez et al., 2021; Kim et al., 2021), and discontinuum mechanics (Shen et al., 2018; Li and Zhao, 2018). The runout analysis models have been used in various capacities to analyze the debris-flow characteristics, such as the flowpath location, volume, velocity, depth, and impact pressure, and determine the performance of barriers.

The empirical runout models are developed from the debris-flow case studies. The empirical models use statistical correlations to estimate the parameters of debris flow based on other variables. For example, the angle of reach ("fahrböschung") was used to estimate the volume of the debris flow (McDougall, 2014). Empirical methods used the volume and the volumetric discharge rate of debris flow to estimate the speed and height of the debris flow (Rickenmann, 1999; Marchi, L. and D'Agostino, 2004). In addition, the impact pressure of debris-flow against a rigid wall has been commonly computed based on the Froude number (Costa, 1984; Hübl and Holzinger, 2003; Tiberghien et al., 2007; Scheidl et al., 2013; and Cui et al., 2015).

Although the empirical model provides a simple and powerful tool, the accuracy is limited by the debris-flow database. The applicability of an empirical model depends on the debris-flow case studies database; for example, an empirical model developed with debris-flow cases from Italy might not be applicable for analyzing debris-flow in South Korea. Furthermore, the empirical models have difficulty in analyzing the performance of an open-type barrier; hence, the empirical models are often limited to analyzing closed-type barriers.

A full 3D simulation model would be required to evaluate the performance of open-type barriers. The 3D simulation was often based on fluid simulation; hence, the runout mass was modeled either as a continuum fluid body based on the continuum mechanics or as spherical particles based on discontinuum methods. The fluid runout mass was governed by Navier-Stokes equations, which are partial differential equations satisfying the dynamic equilibrium of Newton's law of motions. For such dynamic conditions, the explicit analysis methods are commonly used to update the state (location, stress, strain, depth, volume, and velocity) of the fluid over time. Although several 3D fluid simulation softwares (Hirt and Nichols, 1988;

Brinkgreve, 2010; ABAQUS, 2011; Itasca Consulting Group, 2019; Domínguez et al., 2021) are available, they are still considered relatively slow and would be inefficient for analyzing debris-flow propagation.

Therefore, many recently developed runout models utilize the equivalent fluid principle (Hungr, 1995) to model runout materials as a simplified continuum fluid material. Additionally, the depth-averaged Navier-Stokes equations, i.e., St. Venant equations or shallow-water equations (SWE), efficiently simulate the runout simulation with realistic representation (Vreugdenhil, 1994). However, existing runout models do not model the barriers in the simulation; instead, the barriers would be modeled as part of the terrain at which the debris-flow can propagate. Hence, these simulation methods do not account for the loss of kinetic energy at the moment of a collision but indirectly compute the energy loss through friction parameters. Therefore, the SWE-based runout simulations have been used in limitation capacity, approximating the performance of closed-type barriers against debris-flow (Remaitre et al., 2008; Liu et al., 2013; Kwan et al., 2015; Dai et al., 2017; Chen et al., 2019; Cuomo et al., 2019; Choi et al., 2021) (Table 2.1).

Table 2.1 Limitation of current 3D numerical analysis runout models

Name	Model Method	Depth-Averaging	Entrainment	Variation in Rheology	Collision with Rigid Barrier	Reference
FLOW-3D	VOF	X	O	O	O	Kim et al. (2021) Hirt and Nichols (1988)
ABAQUS	FEM, CEL, SPH	X	O	O	O	ABAQUS (2011) Lee and Jeong (2018)
FLOW-R	spreading	O	X	X	X	Horton et al. (2013)
FLO-2D	FDM	O	X	O	X	FLO-2D Software Inc. (2007)
DAN3D	SPH	O	O	O	X	McDougall (2006)
GeoFlow-SPH	SPH	O	O	O	X	Pastor et al. (2009)

Note: VOF = volume of fluid; FEM = finite element method; CEL = coupled Eulerian Lagrangian method; SPH = smoothed particles hydrodynamics; FDM = finite difference method

The SPEC-debris runout simulation method was developed to overcome the limitations of existing runout depth-averaging models through model incorporation of collision physics; therefore, the SPEC-debris model can analyze the performance of even open-type barriers. The purpose of Chapter 2 is to describe the governing equations of the developed SPEC-debris method.

## 2.2 Governing Equations of SPEC-debris

### 2.2.1 Overview

Following the equivalent fluid principle (Hung, 1995), the runout materials, such as snow, water, soil, and rock, are modeled as single-phase “equivalent fluid” with simple internal and basal rheological properties (Savage and Hutter, 1989), similar to the existing depth-averaged continuum mechanics-based runout analysis methods (see section 2.1).

The developed SPEC-debris analysis model is based on depth-averaged Navier-Stokes equations, i.e., shallow-water equations (SWE) or St. Venant equations. In typical debris-flow hazards, the debris mass usually spreads over a large region; therefore, the area of the debris-flow is considered relatively more significant than the debris-flow depth. In such runout mass, the variation in the horizontal pressure gradient along the vertical length is considered small; hence, the horizontal velocity is considered a constant and is computed by integrating the horizontal velocity along with the fluid depth. The vertical integration, equivalent to computing the average horizontal velocity, simplifies the full Navier-Stokes equations to SWE by removing the vertical velocity component (Figure 2.1). In the 3D simulation, only the depth-averaged horizontal velocity for the X and Y directions are considered. Many popular runout simulation methods (McDougall, 2016) have utilized the depth-averaged fluid models to produce reasonably accurate simulation results while increasing the computation efficiency (Hung, 1995; McDougall, 2006; Hung and McDougall, 2009; Christen et al., 2010; Pastor et al., 2014; Mergili et al., 2017).

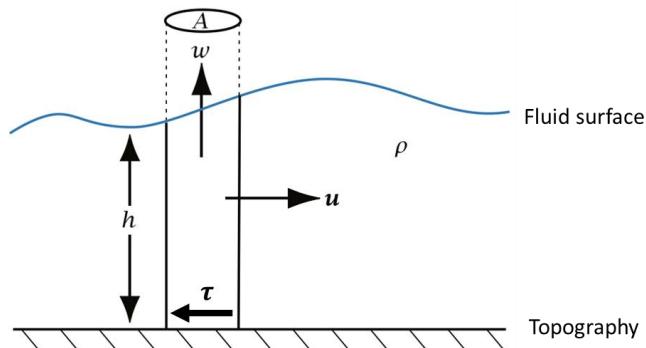


Figure 2.1 Diagram of fluid governed by the St. Venant equations

The SPEC-debris analysis method utilizes the following governing equations to simulate debris-flow propagation and analyze the performance of barriers against debris-flow:

- smoothed particle hydrodynamics (SPH) computing the depth and pressure gradients,
- pathway model computing the movement direction of debris-flow based on the principles of hydrodynamics,
- energy conservation computing the debris-flow velocity and propagation based on the energy conservation, and
- collision mechanism computing the effect of collision based on the linear momentum conservation.

The first letter of these four governing equations was used to create the name of the developed method, the SPEC-debris (**SPH – Pathway – Energy – Collision – debris**-flow) model.

In SPH numerical analysis, the meshless Lagrangian method is utilized to simulate the fluid dynamics as propagation of discrete particles. Similarly, the discrete element method (DEM) often models the debris mass as spherical particles and uses momentum conservation equations to analyze collision during a hydrodynamic simulation (Li and Zhao, 2018). Therefore, the SPEC-debris method utilizes both SPH numerical method and collision mechanism, discretizes the runout mass into  $N$  number of particles with equal volume, and simulates the particle-to-particle interactions.

## 2.2.2 Smoothed particle hydrodynamics (SPH)

### 2.2.2.1 Theory

The smoothed particle hydrodynamics (SPH) is a numerical analysis method that simulates the motion of continuum media, such as fluids. It was originally developed by Gingold and Monaghan (1977) and Lucy (1977) for simulating astrophysical phenomena. The SPH is a meshfree Lagrangian method that follows the motions of continuum media through the motion of discrete elements, which are typically spherical particles.

In the SPH, interactions between the particles are computed with a weighting function, also referred to as a kernel function ( $W$ ), with the range of influence quantified by the smoothing length ( $l$ ). The SPH utilizes an interpolation theory computing a value of a generic function ( $f$ ) in an arbitrary field by integrating the following equation:

$$f(r) = \int f(r') \cdot W(|r - r'|, l) dr' , \quad (2-1)$$

where  $r$  and  $r'$  are the position vectors,  $W(|r - r'|, l)$  is the interpolating kernel function,  $f(r)$  is the unknown quantity at position vector  $r$ , and  $f(r')$  are known quantity at position vector  $r'$ . The kernel function should integrate to unity (1) across an infinite domain:

$$\int W(|r - r'|, l) dr' = 1 . \quad (2-2)$$

However, instead of computing the value of a generic function ( $f$ ) in an infinite domain, the SPH analysis method approximates discrete values from a set of particles located in a nearby region. The Eq. (2-1) is approximated using a Riemann summation over the particles:

$$f(r_i) = \sum_{j=1}^N f(r_j) \cdot W_{ij}(|r_i - r_j|, l) \cdot V_j , \quad (2-3)$$

where  $r_i$  is the position vector of a particle with unknown quantity  $f(r_i)$ ,  $r_j$  are the position vector of other particles with known quantity  $f(r_j)$ ,  $W_{ij}(|r_i - r_j|, l)$  is the kernel function,  $V_j$  is the volume of the particle at position vector  $r_j$ , and  $N$  is the number of neighboring particles influencing the particle at position vector  $r_i$ . Eq. (2-3) is the generalized form of SPH interpolation for three-dimension (3D) fluid flow. This equation can

be adopted for depth-averaged assumption by computing the area of neighboring particles instead of the volume of the particles. Assuming that debris-flow is incompressible (Hung, 1995; McDougall, 2006; Pastor et al., 2014), the following ratio can compute the area of a particle:

$$A = V/h, \quad (2-4)$$

where  $A$  is the particle area,  $h$  is the particle depth, and  $V$  is the particle volume. Therefore, Eq. (2-3) is modified with Eq. (2-4) for the depth-averaged SPH interpolation:

$$f(r_i) = \sum_{j=1}^N f(r_j) \cdot W_{ij}(|r_i - r_j|, l) \cdot \frac{V_j}{h_j}. \quad (2-5)$$

For the sake of clarity, the Eq. (2-5) is simplified as follows:

$$f_i = \sum_{j=1}^N \frac{V_j}{h_j} \cdot f_j \cdot W_{ij}, \quad (2-6)$$

where the subscript  $i$  represents the selected particle, subscript  $j$  represents the surrounding particles neighboring the selected particle  $i$ ,  $f$  is the generic value,  $V$  is the particle volume,  $h$  is the particle depth, and  $W_{ij}$  is the SPH kernel function.

The SPH interpolation scheme can also compute the gradient of a generic function ( $\nabla f$ ) by utilizing a gradient of the kernel function ( $\nabla W$ ):

$$\nabla f_i = \sum_{j=1}^N \frac{V_j}{h_j} \cdot f_j \cdot \nabla W_{ij}. \quad (2-7)$$

### 2.2.2.2 Depth and depth gradient interpolation

The SPEC-debris method utilizes the SPH interpolation method, as defined in Eq. (2-6) and (2-7), to determine the depth ( $h$ ) and gradient of depth ( $\nabla h$ ) at a particular particle location by substituting the generic function ( $f$ ) with particle depth ( $h$ ). With the substitution, the depth of neighboring particles ( $h_j$ ) on the right side of the equations are canceled; hence, providing the following equations:

$$h_i = \sum_{j=1}^N V_j \cdot W_{ij}, \quad (2-8)$$

$$\nabla h_i = \sum_{j=1}^N V_j \cdot \nabla W_{ij}. \quad (2-9)$$

where the subscript  $i$  represents the selected particle, subscript  $j$  represents the particles neighboring the selected particle  $i$ ,  $N$  is the number of neighboring particles,  $h$  is the particle depth,  $V$  is the particle volume,  $W_{ij}$  is the SPH kernel function, and  $\nabla W_{ij}$  is the gradient of the SPH kernel function. Other runout simulation methods incorporating SPH methods, such as DAN3D (Hung and McDougall, 2009) and GeoFlow-SPH (Pastor et al., 2014), use similar methods to compute the depth.

Using the Eq. (2-8), the change of the depth ( $\Delta h$ ) can be computed from the volume change ( $\Delta V$ ):

$$\Delta h_i = \sum_{j=1}^N \Delta V_j \cdot W_{ij} , \quad (2-10)$$

where the subscript  $i$  represents the selected particle, subscript  $j$  represents the particles neighboring the selected particle  $i$ ,  $N$  is the number of neighboring particles,  $\Delta h$  is the change of depth,  $\Delta V$  is the change of neighboring particle volume, and  $W_{ij}$  is the SPH kernel function. The increase in particle height is from entrainment when the runout mass increases by eroding and combing the topsoil. Therefore, an increase in particle height corresponds to a decrease in the layer of topsoil; hence, Eq. (2-10) can be used to compute the amount of eroding depth due to entrainment.

The SPEC-debris method selected the Gaussian kernel function, as recommended by Monaghan (1992). The SPEC-debris model utilizes depth-averaging; therefore, a particle represents a shallow column of debris-flow. Due to the shallow depth, the influence between particle-columns would mainly depend on the horizontal distance. Furthermore, no particles-columns are vertically stacked on top of each other; hence, the vertical distance depends on the 3D terrain. Assuming that the amount of elevation change in the 3D terrain is relatively small compared to the horizontal scales of the debris-flow, the interpolation scheme can be approximated by the 2D form Gaussian kernel function. Other depth-averaged runout models for 3D terrains have utilized 2D SPH interpolation schemes (McDougall, 2006).

The 2D Gaussian kernel function ( $W_{ij}$ ) and the gradient magnitude of the 2D Gaussian kernel function ( $|\nabla W_{ij}|$ ) are defined by the following equations:

$$x_{ij} = (x_i - x_j) ; \quad y_{ij} = (y_i - y_j) ; \quad z_{ij} = (z_i - z_j) ; \quad s_{ij} = \sqrt{x_{ij}^2 + y_{ij}^2 + z_{ij}^2} , \quad (2-11)$$

$$q = s_{ij}/l , \quad (2-12)$$

$$W_{ij} = \begin{cases} \frac{1}{\pi l^2} e^{-q^2}, & q < 2 \\ 0, & q \geq 2 \end{cases} , \quad (2-13)$$

$$|\nabla W_{ij}| = \begin{cases} \frac{2q}{\pi l^2} e^{-q^2}, & q < 2 \\ 0, & q \geq 2 \end{cases} , \quad (2-14)$$

where the subscript  $i$  and  $j$  are a selected particle and neighboring particles surrounding the selected particle;  $x$ ,  $y$ , and  $z$  are the particle XYZ coordinates;  $s_{ij}$  is the distance between particles; and  $l$  is the smoothing length. The Gaussian kernel function has a horizontal asymptote where the kernel value ( $W_{ij}$ ) does not reach zero (0). In order to increase computation efficiency and limit the range of influence between the particles, the kernel value is set to zero (0) when the distance between the particles is twice the smoothing length ( $l$ ), i.e.,  $q \geq 2$ .

In the SPH interpolation method for computing a gradient of a generic function ( $\nabla f$ ), only the magnitude of the gradient is computed. In the SPEC-debris model, the chain rule is used to determine the gradient of depth ( $\nabla h$ ) at X and Y-directions:

$$\left(\frac{\partial h}{\partial x}\right)_i = \sum_{j=1}^N V_j \cdot |\nabla W_{ij}| \cdot \frac{x_{ij}/l}{\sqrt{x_{ij}^2 + y_{ij}^2}}, \quad (2-15)$$

$$\left(\frac{\partial h}{\partial y}\right)_i = \sum_{j=1}^N V_j \cdot |\nabla W_{ij}| \cdot \frac{y_{ij}/l}{\sqrt{x_{ij}^2 + y_{ij}^2}}. \quad (2-16)$$

where the subscript  $i$  and  $j$  are a selected particle and neighboring particles surrounding the selected particle;  $x_{ij}$  and  $y_{ij}$  are the distance between particles;  $N$  is the number of neighboring particles;  $V$  is the particle volume;  $l$  is the smoothing length; and  $|\nabla W_{ij}|$  is the gradient magnitude of the 2D Gaussian kernel function. The depth gradients are used to determine the gradient of lateral hydrodynamic pressure, a key component in the pathway algorithm for determining the movement direction of the particles.

### 2.2.2.3 Adaptive smoothing length

The smoothing length ( $l$ ) is the critical component in the SPH method. The amount of influence that a particle exerts on other particles is within twice the smoothing length. The original SPH formulation was designed for a fixed value of smoothing length ( $l$ ). However, a fixed smoothing length may lead to inaccurate results. If the particles are sparsely distributed, the number of neighboring particles may be insufficient to interpolate the particle depth and depth gradient accurately. Therefore, an adaptive smoothing length method was utilized in the SPEC-debris model.

The SPEC-debris method adopted the adaptive smoothing length developed by Monaghan (1985) and McDougall (2006). The value of smoothing length varies during the calculation based on the average particle area ( $A$ ), which can be estimated based on Eq. (2-4):

$$l = \frac{l_k}{\sqrt{\frac{\sum_i A_i}{N}}} = \frac{l_k}{\sqrt{\frac{\sum_i (h_i/V_i)}{N}}}, \quad (2-17)$$

where  $l$  is the smoothing length,  $l_k$  is the constant for adaptive smoothing length computation,  $A$  is the particle area,  $h$  is the particle depth,  $V$  is the particle volume, and  $N$  is the number of particles. DAN3D has also adopted a similar adaptive smoothing length method and utilized a value of four (4) for the constant  $l_k$  (McDougall, 2006); therefore, the SPEC-debris model also utilized the same value for the constant  $l_k$ .

### 2.2.2.4 Neighboring particles

The SPH interpolation method uses the parameters of the surrounding particles. According to the Gaussian kernel function defined in Eq. (2-13) and (2-14), any particle located further than twice the smoothing length, i.e.,  $q > 2$ , would not influence the SPH interpolation. Therefore, only the neighboring particles, i.e., particles located less than twice the smoothing length, need to be considered for SPH interpolation. Considering only the neighboring particles would increase the computation efficiency, the SPEC-debris model utilizes a KD-tree search-based kNN (k-number of nearest neighbor) algorithm to select only the neighboring particles for SPH interpolation (Bentley, 1975).

The SPEC-debris takes an additional step for selecting the neighboring particles when a barrier or wall is present. The traditional SPH neighboring particle selection method would only consider the radial distance between the particles; however, particles separated by a wall would not influence each other. Hence, the developed model considers the effect of a barrier in SPH interpolation by excluding particles located on the other side of the wall.

As shown in Figure 2.2, when a barrier is modeled in the SPEC-debris analysis method, two regions are generated near the wall. If a barrier is nearby, the developed method will check whether the surrounding particles are within the same region or not. For example, if a particle is located in a different region, it will be excluded from the neighboring particles and not considered for SPH interpolation.

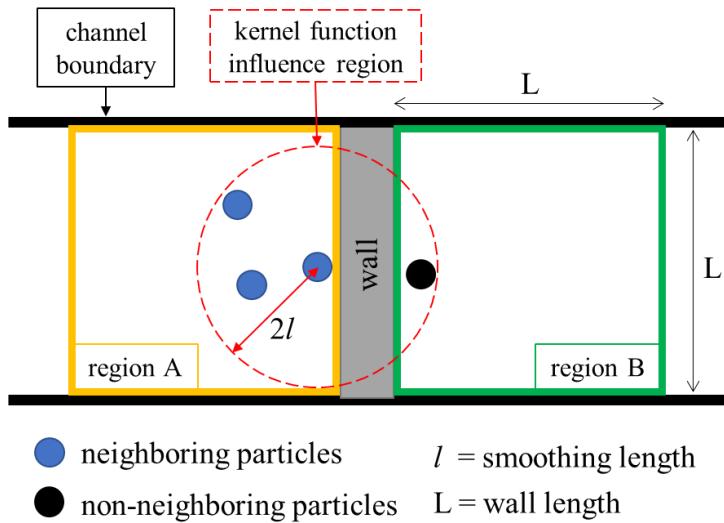


Figure 2.2 Effect of barrier on selecting neighboring particles in SPH interpolation

### 2.2.2.5 Boundary ghost particles

Fluid particles near the edges of the fluid body have fewer neighboring particles than fluid particles in the middle of the fluid body. Hence, the SPH interpolation computes a lower depth value near the edges of the fluid body, which creates tapering of depth at the edges of the fluid body. However, fluid confined inside a box or a flume channel would not have depth tapering at the edges. Therefore, the SPH interpolation method adopted a fluid extension boundary technique with fixed ghost particles by Bouscasse et al. (2013). The ghost particles provide neighboring particles for particles located near such boundaries to interpolate the depth and depth gradient correctly.

The ghost particles are located outside the rectangular DEM boundary in a regular rectangular grid layout, as shown in Figure 2.3. The center-to-center distance between the ghost particles is twice the fluid particles' radius. The ghost particles are unmovable, i.e., velocity is always zero (0). Although the volume of the ghost particles should be updated with SPH interpolation based on the neighboring fluid particles, a simplified method is used where the ghost particle volume is equal to the average volume of the entire fluid particles.

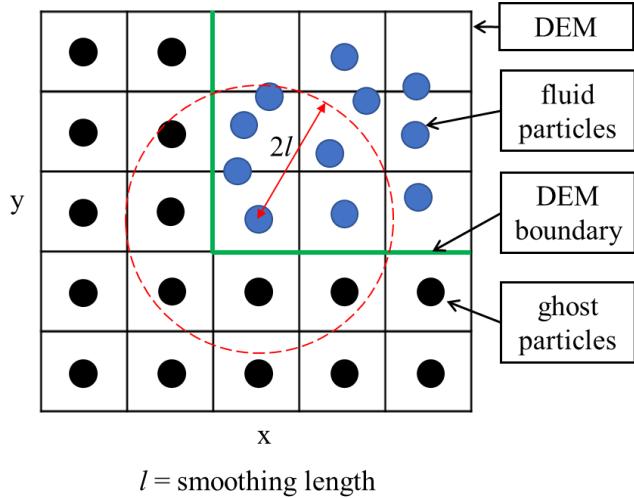


Figure 2.3 Fixed ghost particles outside the DEM boundary for the SPH interpolation

### 2.2.3 Pathway model

The SPEC-debris analysis method determines the direction of particle movement with the pathway algorithm, which considers the following factors:

- gravitational acceleration,
- hydrodynamic pressure gradient applied from the continuum media,
- basal rheological resistance force,
- momentum transfer from entrainment phenomena, and
- inertia, i.e., the movement direction of particles from the previous time step.

#### 2.2.3.1 Gravity

The gravitational acceleration exerted on the particle is based on the terrain specified in the digital elevation model (DEM). Assuming that a small section of the terrain around the particle (henceforth named the local particle terrain) is a plane surface, the gravitational acceleration exerted on the particle will be towards the direction with the largest downward slope. In geology, the maximum local downward inclination angle in a plane is the dip angle, and the direction of the dip angle is the dip direction. Therefore, the magnitude and the direction of gravitational acceleration are defined by the dip angle and dip direction, respectively.

The SPEC-debris method utilized a local linear regression descent method (Hansen et al., 2020) to compute the dip angle and the dip direction in a local n-by-n square neighborhood DEM cell. Assuming that there are no sudden vertical terrain changes in the DEM, selecting an n-by-n square as the local particle terrain allows the following advantages:

- smooth and continuous downwards descent for the particle movement,
- prevents particles from reaching a local minimum created from a small crater or hill,
- less susceptible to error from low DEM resolution, and
- increased computation efficiency.

The odd n-by-n square number is selected to place the particle at the center cell. However, the assumption for selecting local n-by-n square neighborhood DEM cells is invalid in the presence of barriers that introduces a sudden vertical change in the local particle terrain. Therefore, if a barrier or building is present nearby the particle, only the cell containing the particle is selected as the local particle terrain.

The linear gradient of the local terrain ( $\vec{\beta}$ ) is computed by solving the least-squares problem:

$$\min_{\beta_0, \beta_x, \beta_y} \sum_{i=1}^k [z(x_i, y_i) - \beta_0 - \beta_x x_i - \beta_y y_i]^2 , \quad (2-18)$$

where  $x_i$  and  $y_i$  are the X- and Y-coordinates of the grids in the local particle terrain;  $z(x_i, y_i)$  is the elevation at an XY grid location; and  $\beta_x$  and  $\beta_y$  are the linear gradients of the local terrain in the X- and Y-directions. The SPEC-debris method follows the convention where the inclination or dip angle is positive for downwards directions; therefore, the negative value of  $\beta_x$  and  $\beta_y$  from Eq. (2-18) is used.

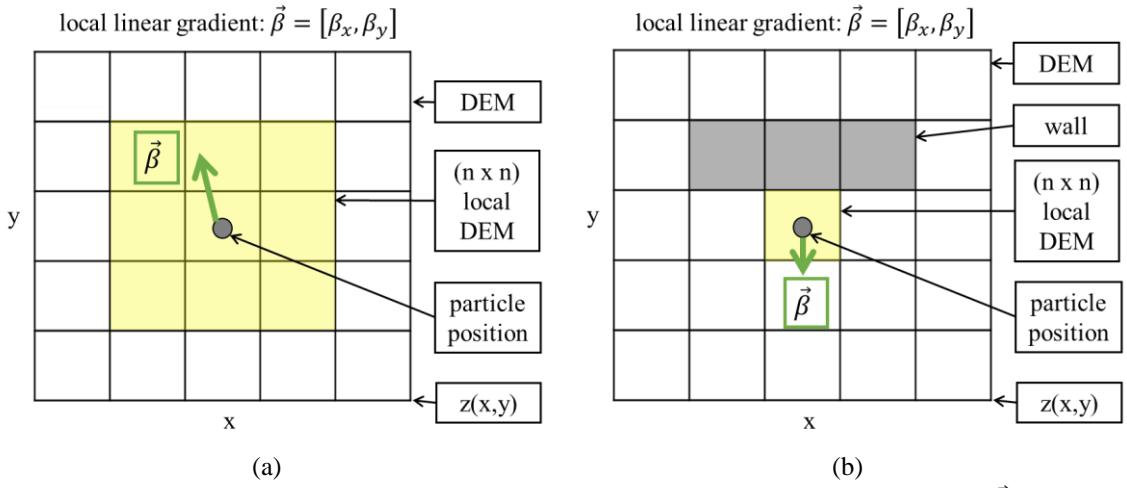


Figure 2.4 Local particle terrain (yellow region) and local linear regression descent ( $\vec{\beta}$ ):

(a) no presence of structures and (b) presence of structures

The dip and dip direction of the local particle terrain are computed from the following equations:

$$\delta = \arccos([1 + \beta_x^2 + \beta_y^2]^{-1/2}) , \quad (2-19)$$

$$\theta_{\delta,P} = \arctan2(\beta_y, \beta_x) , \quad (2-20)$$

where  $\delta$  is the dip angle;  $\theta_{\delta,P}$  is the dip direction in polar coordinates; and  $\beta_x$  and  $\beta_y$  are the linear gradients of the local terrain in the X- and Y-directions. The arctan2 function used in Eq. (2-20) computes the dip direction in the polar coordinates following these rules:

$$\arctan2(\beta_y, \beta_x) = \begin{cases} \arctan(\beta_y/\beta_x) & \text{if } \beta_x > 0, \\ \arctan(\beta_y/\beta_x) + \pi & \text{if } \beta_x < 0 \text{ and } \beta_y \geq 0, \\ \arctan(\beta_y/\beta_x) - \pi & \text{if } \beta_x < 0 \text{ and } \beta_y < 0, \\ +\frac{\pi}{2} & \text{if } \beta_x = 0 \text{ and } \beta_y > 0, \\ -\frac{\pi}{2} & \text{if } \beta_x = 0 \text{ and } \beta_y < 0, \\ \text{undefined} & \text{if } \beta_x = 0 \text{ and } \beta_y = 0. \end{cases} \quad (2-21)$$

Based on the dip and the dip direction, the inclination angle along a particular direction can be computed by:

$$\delta_\theta = \arctan(\tan \delta \cdot \cos|\theta_B - \theta_{\delta,B}|), \quad (2-22)$$

where  $\delta_\theta$  is the inclination angle along the selected direction  $\theta$ ,  $\delta$  is the dip angle,  $\theta_{\delta,B}$  is the dip direction in bearing coordinates, and  $\theta_B$  is the selected direction  $\theta$  in bearing coordinates. The angle conversion between the bearing coordinate system and the polar coordinate system is computed with modular arithmetic:

$$\begin{aligned} \theta_B^\circ &= [450^\circ - \theta_P^\circ] \bmod 360^\circ, \\ \theta_B^c &= [(2.5\pi)^c - \theta_P^c] \bmod (2\pi)^c, \end{aligned} \quad (2-23)$$

where the subscripts B and P refer to bearing and polar coordinate system, respectively; the superscript  $^\circ$  and  $^c$  refers to degrees and radians, respectively; and  $\theta$  is the direction angle.

The apparent dip angle is used to compute the gravitational acceleration in a particular direction for the depth-averaged model:

$$g_\theta = g \sin \delta_\theta, \quad (2-24)$$

where  $g_\theta$  is the gravitational acceleration along a direction  $\theta$ ,  $g$  is the gravitational acceleration in the vertical direction, and  $\delta_\theta$  is the inclination angle along a direction  $\theta$ .

### 2.2.3.2 Hydrodynamic pressure gradient

According to the Navier-Stokes momentum equation, the gravitational force and the stress gradient acting on the fluid body is other driving force of fluid motion. Therefore, the pathway algorithm considers the hydrodynamic pressure exerted on a particle from the surrounding particles to determine the particle movement direction.

In a generalized continuum body in 3D with symmetrical loading conditions, the simulation model solves the complex 3D stress redistribution at each time step. The depth-averaged models developed by Denlinger and Iverson (2004) solve the 3D stress redistribution. However, most depth-averaged models follow the Savage and Hutter (1989) method (henceforth abbreviated as the SH method) and compute the principal normal stresses. The original SH method assumed that the major and minor principal axes are parallel to the

direction of the down-slope and cross-down-slope direction, respectively. Furthermore, the case study model used in the SH method was constructed so that the direction of debris-flow movement is parallel to the down-slope direction.

For irregular 3D terrain, the SPEC-debris method adopted the SH method by assuming that the major and minor principal axes are parallel to the particle movement direction (MD) and cross-movement direction (XMD), similar to the method used by McDougall (2006). The XMD direction angle ( $\theta_{XMD}$ ) are derived from the MD direction angel ( $\theta_{MD}$ ) by:

$$\begin{aligned}\theta_{XMD}^{\circ} &= \theta_{MD}^{\circ} + 90^{\circ}, \\ \theta_{XMD}^c &= \theta_{MD}^c + (\pi/2)^c,\end{aligned}\quad (2-25)$$

where  $\theta$  is the direction angle; the superscript  $^{\circ}$  and  $^c$  refers to degrees and radians, respectively; and the subscript MD and XMD refer to the particle movement direction and cross-movement direction, respectively.

The SH method incorporates Rankine lateral earth pressure coefficients to compute the principal hydrodynamic pressure based on the vertical normal stress. In existing depth-averaged runout models, the vertical normal stress is often computed by gravitational and centripetal acceleration.

$$\sigma_{MD} = k_{MD} \sigma_z, \quad (2-26a)$$

$$\sigma_{XMD} = k_{XMD} \sigma_z, \quad (2-26b)$$

$$\sigma_z = \rho h(g \cos \delta + \kappa u^2), \quad (2-27)$$

where  $\sigma$  is the normal stress,  $k$  is the lateral earth pressure coefficient,  $\rho$  is the density,  $h$  is the depth,  $g$  is the gravitational acceleration,  $\delta$  is the dip angle,  $\kappa$  is the curvature of the local terrain, and  $u$  is the velocity magnitude. The subscripts MD, XMD, and z refer to the particle movement direction, cross-movement direction, and vertical direction, respectively.

In Section 2.2.3.1, the SPEC-debris method assumes that the local particle terrain is a planar surface, as shown in Figure 2.4; hence, the local dip and dip direction can be computed by the local linear regression descent method. On a plane surface, the curvature of the local terrain is zero (0), as the radius of curvature is infinite. Therefore, the Eq. (2-27) can be simplified to:

$$\sigma_z = \rho h g \cos \delta. \quad (2-28)$$

According to the Navier-Stokes momentum equation in the Lagrangian framework, the horizontal acceleration due to the pressure gradient component is the following:

$$a_{\nabla P} = \frac{Du}{Dt} = - \frac{\nabla P}{\rho}, \quad (2-29)$$

where  $a_{\nabla P}$  is the acceleration caused by the pressure gradient,  $u$  is the velocity,  $t$  is the time,  $\nabla P$  is the pressure gradient, and  $\rho$  is the density. By substituting Eq. (2-26) and (2-28) into (2-29), the horizontal acceleration pressure gradients along the MD and XMD direction are computed as:

$$a_{\nabla P, MD} = \frac{Du_{MD}}{Dt} = -g \cos \delta k_{MD} \frac{\partial h}{\partial \ell_{MD}}, \quad (2-30)$$

$$a_{\nabla P, XMD} = \frac{Du_{XMD}}{Dt} = -g \cos \delta k_{XMD} \frac{\partial h}{\partial \ell_{XMD}}, \quad (2-31)$$

where  $a_{\nabla P}$  is the acceleration caused by a pressure gradient,  $u$  is the velocity,  $t$  is the time,  $g$  is the gravitational acceleration,  $\delta$  is the dip angle,  $k$  is the lateral earth pressure coefficient,  $h$  is the depth, and  $\ell$  is length. The subscripts MD and XMD refer to the particle movement direction and cross-movement direction, respectively.

Instead of using the global XY coordinate system, the SH method computes the local pressure gradients in MD and XMD, as shown in Eq. (2-30) and (2-31). Assuming that the particle movement direction MD aligns with the major principal axis, the shear strain term is no longer required. The SPEC-debris method rotates the neighboring particles according to the MD direction angle ( $\theta_{MD}$ ) so that the MD is aligned with the global X-direction. Based on the reoriented local terrain, the strain ( $\varepsilon$ ), lateral earth pressure coefficient ( $k$ ), and the depth gradient ( $\partial h / \partial \ell$ ) are computed.

The incremental principal longitudinal strains ( $\Delta\varepsilon_{MD}, \Delta\varepsilon_{XMD}$ ) determine whether the lateral pressure is in an active or passive state in the MD and XMD. The principal strains are computed by the strain rosette method used for strain measurement from strain gauges rosettes. In DAN3D (McDougall, 2006), a simplified version of the 2D strain rosette method provided the incremental principal longitudinal strains:

$$\Delta\varepsilon_{ij} = \frac{s_{ij}^{t-1} - s_{ij}^t}{s_{ij}^{t-1}}, \quad (2-32)$$

$$\theta_{ij} = \arctan2(\ell_{MD,ij}, \ell_{XMD,ij}), \quad (2-33)$$

$$\begin{aligned} \Delta\varepsilon_{ij} &= \left( \frac{\Delta\varepsilon_{MD} + \Delta\varepsilon_{XMD}}{2} \right) + \left( \frac{\Delta\varepsilon_{MD} - \Delta\varepsilon_{XMD}}{2} \right) \cos(2\theta_{ij}), \\ \Delta\varepsilon_{ij} &= c_\varepsilon + m_\varepsilon \cdot \cos(2\theta_{ij}), \end{aligned} \quad (2-34)$$

where the subscripts  $i$  and  $j$  refer to selected particles and neighboring particles (as defined in Section 2.2.2.4), respectively;  $s_{ij}^{t-1}, s_{ij}^t$  (as defined in Eq. (2-11)) are the distance between the particles  $i$  and  $j$  at the old previous time step and current time step, respectively;  $\Delta\varepsilon_{ij}$  is the incremental strain between particles  $i$  and  $j$ ;  $\ell_{MD,ij}, \ell_{XMD,ij}$  are the distance between particles  $i$  and  $j$  in MD and XMD direction;  $\theta_{ij}$  is the polar angle coordinate between particles  $i$  and  $j$ ;  $\Delta\varepsilon_{MD}, \Delta\varepsilon_{XMD}$  are the incremental principal longitudinal strains in MD and XMD, respectively; and  $c_\varepsilon, m_\varepsilon$  are the vertical axis intercept and the slope.

The SPEC-debris computes the  $\Delta\varepsilon_{ij}$  and  $\theta_{ij}$  between the selected particle and the neighboring particles, then uses least-square regression analysis to determine the intercept and the slope of the Eq. (2-34). The incremental principal strains in the MD and XMD are computed as:

$$\Delta\varepsilon_{MD} = c_\varepsilon + m_\varepsilon, \quad (2-35)$$

$$\Delta\varepsilon_{XMD} = c_\varepsilon - m_\varepsilon. \quad (2-36)$$

When the incremental principal strain is a positive number, the fluid body is in a passive stress state. The fluid body is in an active state for incremental principal strain at zero (0) or a negative number.

The SH method computes the active and passive Rankine lateral earth pressure coefficients ( $k_a, k_p$ ) in the MD and XMD based on simple internal and basal rheological properties. According to the equivalent fluid principle, the internal and basal rheological properties are based on the internal friction angle ( $\phi_i$ ) and the basal friction angle ( $\phi_b$ ). The following section will discuss the basal rheological properties and basal shear resistance stress. For general application, the basal friction angle ( $\phi_b$ ) is computed as the ratio between vertical normal stress ( $\sigma_z$ ) and basal shear resistance stress ( $\tau_b$ ):

$$\phi_b = \arctan \left| \frac{\tau_b}{\sigma_z} \right|. \quad (2-37)$$

The SH method determines the lateral earth pressure coefficients in MD and XMD ( $k_{MD}, k_{XMD}$ ) based on the incremental principal strains ( $\Delta\varepsilon_{MD}, \Delta\varepsilon_{XMD}$ ) and rheological friction ( $\phi_i, \phi_b$ ):

$$k_{MD} = \begin{cases} k_p = 2 \sec^2 \phi_i \left[ 1 + \sqrt{1 - (\cos^2 \phi_i / \cos^2 \phi_b)} \right] - 1 & \text{if } \phi_b \leq \phi_i \text{ and } \Delta\varepsilon_{MD} > 0, \\ k_a = 2 \sec^2 \phi_i \left[ 1 - \sqrt{1 - (\cos^2 \phi_i / \cos^2 \phi_b)} \right] - 1 & \text{if } \phi_b \leq \phi_i \text{ and } \Delta\varepsilon_{MD} \leq 0, \\ k = 2 \sec^2 \phi_i - 1 & \text{if } \phi_b > \phi_i, \end{cases} \quad (2-38)$$

$$k_{XMD} = \begin{cases} k_p = \frac{1}{2} \left( k_{MD} + 1 + \sqrt{(k_{MD} - 1)^2 + 4 \tan^2 \phi_b} \right) & \text{if } \phi_b \leq \phi_i \text{ and } \Delta\varepsilon_{XMD} > 0, \\ k_a = \frac{1}{2} \left( k_{MD} + 1 - \sqrt{(k_{MD} - 1)^2 + 4 \tan^2 \phi_b} \right) & \text{if } \phi_b \leq \phi_i \text{ and } \Delta\varepsilon_{XMD} \leq 0, \\ k_p = 1/(1 - \sin \phi_i) & \text{if } \phi_b > \phi_i \text{ and } \Delta\varepsilon_{XMD} > 0, \\ k_a = 1/(1 + \sin \phi_i) & \text{if } \phi_b > \phi_i \text{ and } \Delta\varepsilon_{XMD} \leq 0, \end{cases} \quad (2-39)$$

where  $\phi_i, \phi_b$  are the internal and basal rheological friction angle;  $\Delta\varepsilon_{MD}, \Delta\varepsilon_{XMD}$  are the incremental principal strains in MD and XMD; and  $k$  is the lateral pressure coefficient with the subscripts  $a, p, MD$ , and  $XMD$  referring to the active state, passive state, particle movement direction, and particle cross-movement direction, respectively. At the beginning of the simulation, the runout material is assumed to be in a hydrostatic fluid state; therefore,  $k_{MD} = k_{XMD} = 1$  at time step = 0.

The SH method instantly transitions between active and passive stress states depending on the incremental principal strains. For example, some runout simulation models utilize an increment approach to increase or decrease the lateral pressure coefficient based on a dimensionless stiffness coefficient (Hung, 1995; Denlinger and Iverson, 2004; Hung and McDougall, 2009). However, most runout models are insensitive to the amount of stiffness coefficient and follow the instantaneous stress response to strain in the SH method. Such an assumption may be valid as the deformation is extremely fast and the magnitude is large during a runout, which computes a considerably large strain value and significant stress changes.

### 2.2.3.3 Basal rheological resistance

The SPEC-debris method used the Voellmy (1955) rheological model to describe the basal rheology properties. Although developed for snow avalanche simulation, the Voellmy model has been adopted to simulate debris-flow due to a similar range in velocity and shape of deposits (Körner, 1976). The Voellmy rheology combines the basal resistance due to basal friction and turbulence flow:

$$\tau_b = \sigma_z f_b + \rho g u^2 f_t, \quad (2-40)$$

where  $\tau_b$  is the basal rheological resistance shear stress,  $\sigma_z$  is the vertical normal stress,  $\rho$  is the density,  $g$  is the gravitational acceleration,  $u$  is the velocity,  $f_b$  is the base resistance coefficient, and  $f_t$  is the turbulence resistance coefficient. In the original Voellmy formulation, the turbulence coefficient ( $\xi$ ) is used; however, converting the Voellmy rheology to simple Coulomb or frictional rheology requires assigning an infinite to the turbulence coefficient ( $\xi$ ). Therefore, the turbulence resistance coefficient ( $f_t = 1/\xi$ ) is used in the SPEC-debris method. Eq. (2-40) is used to compute the basal friction angle by Eq. (2-37).

By substituting Eq. (2-4) and (2-28), the deceleration caused by the Voellmy basal resistance is:

$$a_{b,MD} = g \left( f_b \cos \delta + f_t \frac{u^2}{h} \right), \quad (2-41)$$

where  $a_{b,MD}$  is the deceleration from basal rheological resistance in the particle movement direction (MD),  $g$  is the gravitational acceleration,  $\delta$  is the dip angle,  $h$  is the depth,  $u$  is the velocity,  $f_b$  is the base friction coefficient, and  $f_t$  is the turbulence coefficient. Eq. (2-41) implies that the depth should be a non-zero positive number not to cause a mathematical error. As the velocity magnitude direction is aligned to the particle movement direction (MD), no resistance is assumed in the particle cross-movement direction (XMD).

### 2.2.3.4 Entrainment momentum transfer

The entrainment phenomenon is when the propagation of runout material erodes the top layer of the ground surface and increases the runout volume by combining with the eroded topsoil. The eroded mass combines with the flowing runout mass during the entrainment phenomenon; therefore, an inelastic momentum transfer has occurred. The conservation of linear momentum states that the velocity must decrease corresponding to the amount of mass gained. Therefore, entrainment imposes deceleration on the flowing runout mass. Runout simulation accounts for the deceleration from entrainment momentum transfer in their momentum or acceleration equations (Hung and McDougall, 2009; Pastor et al., 2014). The SPEC-debris method adopted the deceleration from entrainment by McDougall (2006):

$$a_{en,MD} = E_s u^2, \quad (2-42)$$

where  $a_{en,MD}$  is the deceleration from entrainment momentum transfer in the particle movement direction (MD),  $E_s$  is the growth rate, and  $u$  is the velocity. The deceleration from entrainment momentum transfer is applied in the particle movement direction (MD) as the velocity magnitude direction aligns with the MD.

### 2.2.3.5 Pathway algorithm

The pathway algorithm combines the acceleration from gravity, pressure gradient, basal resistance, momentum transfer, and the inertial direction from current velocity to compute the direction of particle movement in the next time step. The following describes the step-by-step computation procedure of the pathway algorithm:

- (1) compute the direction angle of particle movement direction (MD) from the particle velocity in X and Y-directions:

$$\theta_{MD,P} = \begin{cases} \arctan2(u_y, u_x), & u > 0, \\ \theta_{\delta,P}, & u = 0 \text{ and time step} = 0, \end{cases} \quad (2-43)$$

where  $\theta_{MD,P}$  is the direction angle of the particle movement direction (MD) in the polar coordinates system;  $u$  is the velocity magnitude;  $u_x, u_y$  are the velocity in X and Y-direction; and  $\theta_{\delta,P}$  is the dip direction angle. At time step = 0 with stationary velocity, use the dip direction as the MD,

- (2) compute the direction of particle cross-movement direction (XMD) ( $\theta_{XMD,P}$ ) by Eq. (2-25),
- (3) compute the dip ( $\delta$ ) and dip direction ( $\theta_\delta$ ) of the n-by-n local particle terrain by Eq. (2-19) and (2-20),
- (4) compute the apparent dip angle for MD and XMD ( $\delta_{MD}, \delta_{XMD}$ ) by Eq. (2-22) and (2-23),
- (5) compute the gravitational acceleration towards MD and XMD ( $g_{MD}, g_{XMD}$ ) by Eq. (2-24),
- (6) compute the simplified vertical normal stress by Eq. (2-28),
- (7) reorient the neighboring particle positions based on the MD,
- (8) using the reoriented neighboring particle positions, compute the depth gradient ( $\partial h / \partial \ell_{MD}$ ,  $\partial h / \partial \ell_{XMD}$ ) along the MD and XMD with SPH interpolation scheme by Eq. (2-15) and (2-16),
- (9) using the reoriented neighboring particle positions, compute the incremental principal strain ( $\Delta \varepsilon_{MD}, \Delta \varepsilon_{XMD}$ ) along the MD and XMD by Eq. (2-32) to (2-36),
- (10) compute the lateral earth pressure coefficient in MD and XMD ( $k_{MD}, k_{XMD}$ ) by Eq. (2-37), (2-38), (2-39) and (2-40),
- (11) compute the acceleration due to pressure gradient in MD and XMD ( $a_{VP,MD}, a_{VP,XMD}$ ) by Eq. (2-30) and (2-31),
- (12) compute the deceleration due to basal resistance in MD ( $a_{b,MD}$ ) by Eq. (2-41),
- (13) compute the deceleration due to entrainment momentum transfer in MD ( $a_{en,MD}$ ) by Eq. (2-42),
- (14) compute the total acceleration in MD and XMD ( $a_{MD}, a_{XMD}$ ):

$$a_{MD} = g_{MD} + a_{VP,MD} - a_{b,MD} - a_{en,MD}, \quad (2-44)$$

$$a_{XMD} = g_{XMD} + a_{VP,XMD}, \quad (2-45)$$

where  $g$  is the gravitational acceleration,  $a_{VP}$  is the acceleration from pressure gradient,  $a_b$  is the acceleration from basal resistance, and  $a_{en}$  is the acceleration from entrainment momentum transfer,

(15) compute the gradients of movement exerted on the particle in the global X and Y-directions ( $\nabla_x, \nabla_y$ ) based on the acceleration and inertia represented by the velocity:

$$\nabla_x = u_x + (a_{MD} \cos \theta_{MD,P} + a_{XMD} \cos \theta_{XMD,P})\Delta t, \quad (2-46)$$

$$\nabla_y = u_y + (a_{MD} \sin \theta_{MD,P} + a_{XMD} \sin \theta_{XMD,P})\Delta t, \quad (2-47)$$

where  $\nabla_x, \nabla_y$  are the gradients of particle movement in X- and Y-directions;  $u_x, u_y$  are the velocity in X and Y-directions;  $a_{MD}, a_{XMD}$  are the total acceleration in MD and XMD;  $\theta_{MD,P}, \theta_{XMD,P}$  are the direction angle in polar coordinates for MD and XMD; and  $\Delta t$  is the time interval between the time step,

(16) compute the new direction of particle travels in the next time step, i.e., MD at the next time step:

$$\theta_V = \arctan2(\nabla_y, \nabla_x), \quad (2-48)$$

where  $\theta_V$  is the new particle direction angle; and  $\nabla_x, \nabla_y$  are the gradients of particle movement in X- and Y-directions.

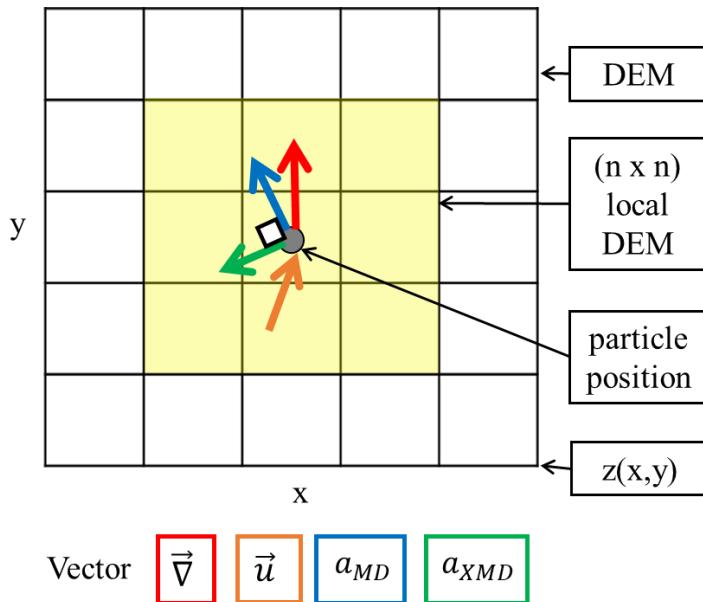


Figure 2.5 Particle movement gradients ( $\vec{\nabla}$ ) computed by the pathway algorithm

The particle movement gradients ( $\nabla_x, \nabla_y$ ) computed from the pathway algorithm are equivalent to computing the particle velocity of the next time step. However, utilizing the pathway algorithm to compute the new particle velocity results in “particle penetration” (Monaghan, 1989), at which the particle velocity suddenly increases exponentially near a collision or hydraulic jump. The “particle penetration” causes instability in simulation and produces inaccurate results.

In runout methods with particles, velocity smoothing ensures that the new particle velocity matches the average neighboring particle velocity (Monaghan, 1992; McDougall, 2006). However, a sudden change in particle velocity is expected when fluid collides against a barrier. Hence, such a velocity smoothing method would be inappropriate during collisions as the particle velocity would be different from the average neighboring particle velocity.

The SPEC-debris method uses the energy conservation principle to compute new particle speed, i.e., velocity magnitude, for collision and non-collision situations. The pathway algorithm provides the new particle direction angle ( $\theta_v$ ). The energy conservation governing equation will be described in the next section.

#### 2.2.4 Energy conservation principle

The SPEC-debris method computes the new particle velocity magnitude based on the principle of energy conservation where the total sum of kinetic energy (KE), gravitational potential energy (GPE), work done by a force exerted from pressure gradient, and work done from friction remains constant over time step, i.e.,  $\Delta E = 0$ . The generalized formulation for the energy conservation model for runout simulation is the following:

$$E_k^t = E_k^{(t-1)} - \Delta E_g^t - \Delta E_p^t - \Delta E_f^t, \quad (2-49)$$

where  $E$  is the energy;  $\Delta$  symbolizes change; superscripts  $t, (t-1)$  refer to current time step and previous time step; and subscripts  $k, g, p$ , and  $f$  refer to kinetic energy, gravitational potential energy, pressure, and work done from friction.

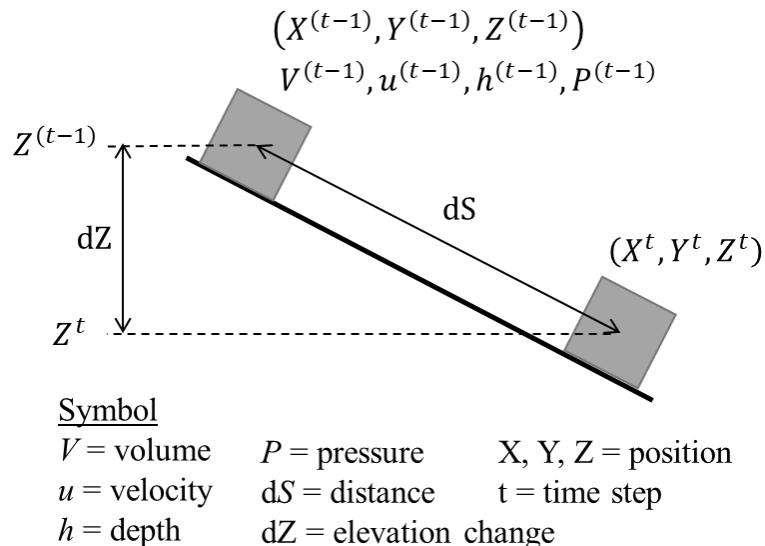


Figure 2.6 General energy conservation model for runout analysis

The SPEC-debris method assumes that the movement of particles between time step occurs within the local particle terrain (see Section 2.2.3.1); therefore, the particle travels in a linear plane. Therefore, the traveled distance is assumed linear and computed by Eq. (2-11).

##### 2.2.4.1 Entrainment

The SPEC-debris energy conservation model incorporates the effect of volume change between the initial and the final state due to entrainment phenomena. Assuming the density remains constant as the SPEC-debris assumes incompressible flow, the volume change leads to mass change. The developed model can assign different masses for initial and final states. A generalized entrainment model is the following:

$$V^t = V^{(t-1)} E(dS), \quad (2-50)$$

where  $V^t$  is the volume at the current time step,  $V^{(t-1)}$  is the volume at the previous time step, and  $E(dS)$  is the entrainment model function.

The SPEC-debris method models two types of entrainment model functions: constant growth defined as Eq. (2-51) and exponential growth defined as Eq. (2-52), which is based on Hungr (1995).

$$E(dS) = E_s dS, \quad (2-51)$$

$$E(dS) = \exp[E_s dS], \quad (2-52)$$

where  $E(dS)$  is the entrainment model function,  $E_s$  is the empirically-derived growth rate per distance traveled, and  $dS$  is the distance between the current and previous time step.

When the particle collides or travels along a barrier wall, the SPEC-debris method assumes no entrainment occurs as no topsoil is eroded; therefore, the entrainment growth rate ( $E_s$ ) is at zero (0). The particle volumes do not change when the collision occurs or the particle travels on the wall surface.

#### 2.2.4.2 Kinetic energy

The kinetic energy is defined by the mass and the velocity. The particle mass is derived from the product of density and particle volume. The kinetic energy at the initial and final states are the followings:

$$E_k^{(t-1)} = \frac{1}{2} \rho V^{(t-1)} (u^{(t-1)})^2, \quad (2-53)$$

$$E_k^t = \frac{1}{2} \rho V^t (u^t)^2 = \frac{1}{2} \rho E(dS) V^{(t-1)} (u^t)^2, \quad (2-54)$$

where  $E_k$  is the kinetic energy;  $\rho$  is the density;  $V$  is the volume;  $E(dS)$  is the entrainment model function;  $u$  is the velocity; and superscripts  $t, (t - 1)$  refer to current and previous time steps, respectively.

#### 2.3.4.3 Gravitational potential energy

The potential energy due to gravity is defined by the mass and the vertical distance above a datum:

$$\Delta E_g^t = -m_{av} g [Z^t - Z^{t-1}] \quad (2-55)$$

where  $\Delta E_g$  is the change in gravitational potential energy;  $g$  is the gravitational acceleration;  $m_{av}$  is the average mass between the current and previous time step;  $Z$  is the elevation; and superscripts  $t, (t - 1)$  refer to the current and previous time step, respectively.

Using the entrainment model defined by Eq. (2-50), the average mass is computed as:

$$m_{av} = \frac{1}{2} [m^t + m^{(t-1)}] = \frac{1}{2} \rho [V^t + V^{(t-1)}] = \frac{1}{2} \rho V^{(t-1)} [E(dS) + 1] \quad (2-56)$$

Where  $m$  is the mass,  $\rho$  is the density,  $V$  is the volume, and  $E(dS)$  is the entrainment model function.

#### 2.2.4.4 Pressure potential energy

As described in Section 2.2.3.2, the hydrodynamic pressure gradient provides a potential energy that drives the fluid from a higher to a lower pressure region. By the definition of work done, the distance traveled due to the force exerted on the fluid body from the pressure gradient provides the pressure potential energy. The pressure force is exerted along the direction of the fluid body expansion; therefore, the active lateral pressure coefficient in the movement direction (MD) by Rankine is used to compute the magnitude of the lateral pressure exerted on the fluid body.

Based on the acceleration due to the pressure gradient, Newton's 2<sup>nd</sup> law of motion and the definition of work done to provide the potential energy due to the pressure change:

$$\Delta E_p = \frac{\Delta P}{\rho} \approx \frac{\nabla P dS}{\rho} = \frac{dS}{\rho} \cdot \frac{(\rho V \cdot a_{\nabla P})}{(V/h)} = -g \cos \delta k_a (h \cdot \nabla h) dS \quad (2-57)$$

where  $\Delta E_p$  is the potential energy from pressure change ( $\Delta P$ ),  $\rho$  is the density,  $V$  is the volume,  $g$  is the gravitational acceleration,  $\delta$  is the dip angle,  $k_a$  is the Rankine's active lateral pressure coefficient,  $h$  is the depth,  $\nabla h$  is the depth gradient, and  $dS$  is the distance traveled.

#### 2.2.4.5 Rheological frictional energy loss

As described in Section 2.2.3.3, the Voellmy rheological model is used to simulate the basal resistance in the SPEC-debris method. The frictional force of Voellmy rheology is defined by substituting Eq. (2-41) into the acceleration of Newton's 2<sup>nd</sup> law of motion:

$$F_f = \rho V \cdot g \left( f_b \cos \delta + f_t \frac{u^2}{h} \right) \quad (2-58)$$

where  $F_f$  is the frictional force,  $\rho$  is the density,  $V$  is the volume,  $g$  is the gravitational acceleration,  $\delta$  is the dip angle,  $h$  is the depth,  $u$  is the velocity,  $f_b$  is the base friction coefficient, and  $f_t$  is the turbulence coefficient. As the particle propagates, the volume ( $V$ ), depth ( $h$ ), and velocity ( $u$ ) of the particle change between the current and previous time steps; hence, the frictional force ( $F_f$ ) changes based on the amount of distance traveled between the current and previous time steps, as shown in Figure 2.7.

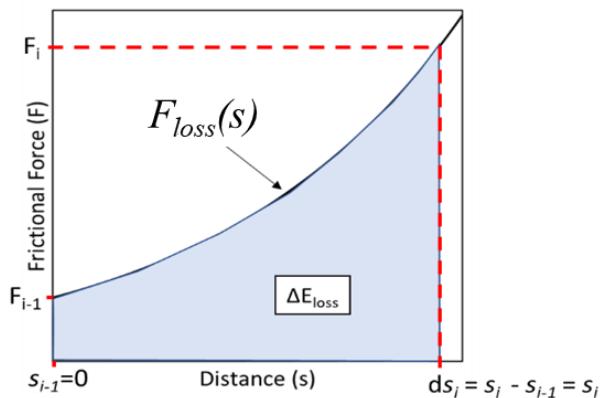


Figure 2.7 Energy loss from the variable frictional force

The entrainment model defined in Eq. (2-50) accounts for the increase in the particle volume, and an average particle velocity between the current and previous time steps is used. At the current time step, the depth of the particle is an unknown quantity solved by the SPH interpolation and difficult to compute using empirical methods. The SPEC-debris method assumed that the incremental change in depth is relatively small; hence, the average particle depth between the current and previous time steps is considered the same as the particle depth at the previous time step.

The work done from the frictional force is computed by integrating the variable frictional force over a distance traveled:

$$\Delta E_{loss} = \int_0^{ds} F_f(S) dS = \rho V^{(t-1)} g \left[ f_b \cos \delta + f_t \frac{(u^t + u^{(t-1)})^2}{4h^{t-1}} \right] \int_0^{ds} E(dS) dS, \quad (2-59)$$

where  $F_f(s)$  is the variable frictional force,  $\rho$  is the density,  $V$  is the volume,  $g$  is the gravitational acceleration,  $\delta$  is the dip angle,  $h$  is the depth,  $u$  is the velocity,  $f_b$  is the base friction coefficient,  $f_t$  is the turbulence coefficient,  $E(dS)$  is the entrainment model function, and superscripts  $t$  and  $(t - 1)$  refer to the current time step and previous time steps.

The energy loss from Voellmy rheology frictional force is only considered as the particle is traveling along the soil surface. Therefore, the frictional force is assumed to be zero as the particle is at the free-fall state.

#### 2.2.4.6 Energy conservation model

The SPEC-debris method combines the kinetic energy, gravitational potential energy, and rheological frictional energy loss into the energy conservation model defined in Eq. (2-49). The following generalized energy conservation model, which incorporates the entrainment phenomena, is used in the developed runout simulation to compute the velocity magnitude:

$$E(dS)(u^t)^2 = (u^{(t-1)})^2 - g[1 + E(dS)] [Z^t - Z^{(t-1)}] \\ + \frac{2}{\rho V^{(t-1)}} g \cos \delta k_a (h^{(t-1)} \cdot \nabla h^{(t-1)}) dS \\ - 2g \left[ f_b \cos \delta + f_t \frac{(u^t + u^{(t-1)})^2}{4h^{(t-1)}} \right] \int_0^{ds_i} E(dS) dS, \quad (2-60)$$

where  $E(dS)$  is the entrainment model function,  $dS$  is the distance between current and previous time step,  $u$  is the velocity,  $h$  is the depth,  $g$  is the gravitational acceleration,  $Z$  is the elevation,  $\delta$  is the dip angle,  $k_a$  is Rankine's active lateral pressure coefficient,  $\nabla h$  is the depth gradient,  $f_b$  is the base friction coefficient, and  $f_t$  is the turbulence coefficient. The superscripts  $t$  and  $(t - 1)$  refer to the current and previous time steps.

If the turbulence coefficient ( $f_t$ ) is a non-zero value, the velocity in the current time step ( $u^t$ ) is solved with the quadratic formula. The following lists the rearranged form of the energy conservation model based on the entrainment model function:

- constant volume, i.e., no entrainment, with  $E(dS) = 1$ ,

$$(u^t)^2 \left\{ 1 + dS \frac{f_t g}{2h^{(t-1)}} \right\} + u^t \left\{ dS \frac{f_t g u^{(t-1)}}{h^{(t-1)}} \right\} + \left\{ \begin{aligned} & 2g[z_i - z_{i-1}] + 2gdS \left[ f_b \cos \delta + f_t \frac{(u^{(t-1)})^2}{4h^{(t-1)}} \right] \\ & - \frac{2}{\rho V^{(t-1)}} g \cos \delta k_a (\mathbf{h} \cdot \nabla \mathbf{h}) dS - (u^{(t-1)})^2 \end{aligned} \right\} = 0, \quad (2-61)$$

- constant growth, as defined by Eq. (2-51), and

$$(u^t)^2 \left\{ E_s dS + E_s (dS)^2 \frac{f_t g}{4h^{(t-1)}} \right\} + u^t \left\{ E_s (dS)^2 \frac{f_t g u^{(t-1)}}{2h^{(t-1)}} \right\} + \left\{ \begin{aligned} & g[1 + E_s dS][z_i - z_{i-1}] + gE_s (dS)^2 \left[ f_b \cos \delta + f_t \frac{(u^{(t-1)})^2}{4h^{(t-1)}} \right] \\ & - \frac{2}{\rho V^{(t-1)}} g \cos \delta k_a (\mathbf{h} \cdot \nabla \mathbf{h}) dS - (u^{(t-1)})^2 \end{aligned} \right\} = 0, \quad (2-62)$$

- Hungr's exponential growth, as defined by Eq. (2-52),

$$(u^t)^2 \left\{ e^{E_s dS} + \frac{[e^{E_s dS} - 1]}{E_s} \cdot \frac{f_t g}{2h^{(t-1)}} \right\} + u^t \left\{ \frac{[e^{E_s dS} - 1]}{E_s} \cdot \frac{f_t g u^{(t-1)}}{h^{(t-1)}} \right\} + \left\{ \begin{aligned} & g[1 + e^{E_s dS}][z_i - z_{i-1}] + 2g \frac{[e^{E_s dS} - 1]}{E_s} \left[ f_b \cos \delta + f_t \frac{(u^{(t-1)})^2}{4h^{(t-1)}} \right] \\ & - \frac{2}{\rho V^{(t-1)}} g \cos \delta k_a (\mathbf{h} \cdot \nabla \mathbf{h}) dS - (u^{(t-1)})^2 \end{aligned} \right\} = 0, \quad (2-63)$$

where  $E_s$  is the empirically-derived growth rate per distance traveled, and  $dS$  is the distance between the current and previous time steps.

When a particle travels along a steep slope close to around  $90^\circ$ , the particle is considered to be airborne. When airborne, only the kinetic and potential energies compute the particle velocity. Therefore, the particle in an airborne state traveling in the vertical direction, the energy conservation model simplifies to the following equation:

$$u^t = \sqrt{(u^{(t-1)})^2 - 2g [Z^t - Z^{(t-1)}]}. \quad (2-64)$$

#### 2.2.4.7 Particle displacement without wall collision

The SPEC-debris method utilizes the leapfrog method to compute the particle displacement. The leapfrog method is a second-order numerical procedure for solving explicit simulations. It is considered simple to implement; therefore, it has been widely adopted in many explicit numerical analysis methods. The SPEC-debris method updates the particle position based on the following equation:

$$\begin{bmatrix} X^t \\ Y^t \end{bmatrix} = \begin{bmatrix} X^{t-1} \\ Y^{t-1} \end{bmatrix} + \frac{\Delta t}{2} \begin{bmatrix} u_X^{t-1} + u^t \cos \theta_V^t \\ u_Y^{t-1} + u^t \sin \theta_V^t \end{bmatrix}, \quad (2-65)$$

where  $X, Y$  are the particle positions in global XY-coordinates;  $\Delta t$  is the time interval between the current and previous time step;  $u_X, u_Y$  are the velocities in X- and Y-direction;  $u$  is the velocity magnitude;  $\theta_V$  is the direction angle; and the superscripts  $t$  and  $(t - 1)$  refer to the current and previous time steps.

In order to produce the computed particle position at the current time step ( $X^t, Y^t$ ), the energy conservation model is used. However, the energy conservation model requires the current particle position is required to solve the velocity at the current time step ( $u_X^t, u_Y^t$ ). Therefore, the particle's position and velocity in the leapfrog method are interdependent variables requiring a numerical converge approach. The SPEC-debris method utilizes the following iterative procedure to converge the particle position and velocity:

- (1) from the pathway algorithm, compute particle movement gradients ( $\nabla_x, \nabla_y$ ) and direction angle ( $\theta_V$ ) with Eq. (2-46), (2-47), and (2-48),
- (2) compute the magnitude of the particle movement gradient ( $|\nabla|$ ):

$$|\nabla| = \sqrt{\nabla_x^2 + \nabla_y^2}, \quad (2-66)$$

- (3) use the  $|\nabla|$  as the initial guess for particle velocity of the current time step to compute the particle position with Eq. (2-65). Then compute the new particle velocity with the energy conservation model ( $u_{EC}^t$ ) with Eq. (2-60),
- (4) if  $u_{EC}^t \geq |\nabla|$ , assign the  $u_{EC}^t$  as the particle velocity of the current time, i.e.,  $u^t = u_{EC}^t$
- (5) if  $u_{EC}^t < |\nabla|$ , use the  $u_{EC}^t$  as the next guess for particle velocity of current time step and repeat steps (3) and (5) until either the  $u_{EC}^t$  value converges, or the number of iterations exceeds the iteration limit.

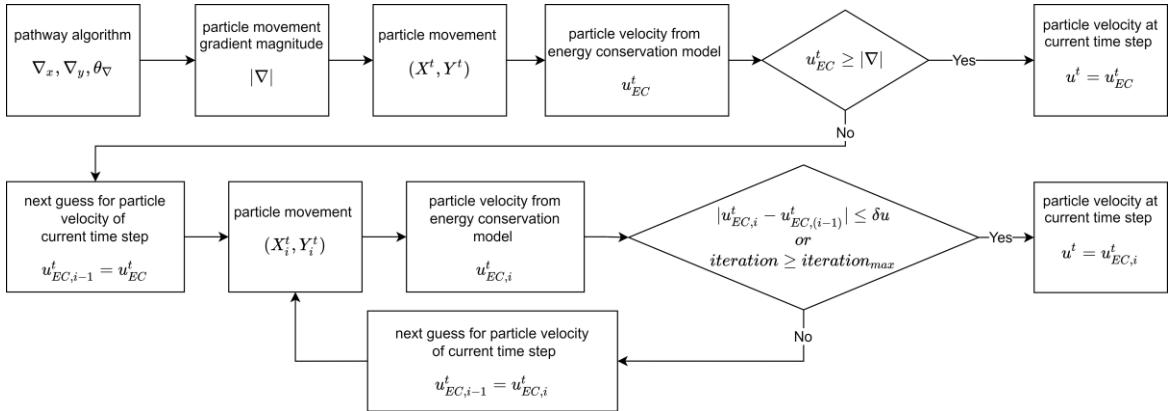


Figure 2.8 Procedure for particle velocity and position at current time step

The velocity convergence at step (5) is when the difference between the guessed and computed velocities is less than or equal to 0.1 m/s. Typically, the convergence usually occurred within three iterations; therefore, for computation efficiency, the maximum iteration is assigned as three (3).

#### 2.2.4.8 Particle displacement along the wall

When a particle collides against a barrier or a building, the particle vertically travels along the wall. Assuming that the particle is airborne and only subjected to the gravitational acceleration, the following SUVAT equation is used to compute the displacement on the vertical axis:

$$Z^t = Z^{t-1} + u^{t-1}\Delta t - \frac{1}{2}g(\Delta t)^2, \quad (2-67)$$

where  $Z$  is the vertical position of the particle,  $\Delta t$  is the time interval between the current and previous time steps,  $u$  is the velocity,  $g$  is the gravitational acceleration, and the superscripts  $t$  and  $(t - 1)$  refer to the current time step and previous time steps. The positive value is assigned to the velocity moving in the upward direction. As the gravity always acts vertically downwards, a negative sign is written for the acceleration component of the equation.

#### 2.2.5 Collision

The SPEC-debris method distinguishes itself from other depth-averaged runout simulation methods by incorporating energy loss from collisions. The method computes the loss of kinetic energy, i.e., velocity, due to colliding against a barrier and turbulent flow through the principle of linear momentum conservation.

The particles are considered hard, undeformed spheres, and the walls are modeled as rigid, stationary objects. The SPEC-debris method utilizes the coefficient of restitution (COR) value to compute the particle velocity after a collision. The COR is defined as:

$$\text{COR} = \frac{\text{relative velocity after collision}}{\text{relative velocity before collision}} = \frac{|u_f|}{|u_0|}, \quad (2-68)$$

$$0 \text{ (perfectly inelastic collision)} \leq \text{COR} \leq 1 \text{ (perfectly elastic collision).}$$

where  $u_0$  is the velocity before collision and  $u_f$  is the velocity after collision. The COR value can vary between zero (0) and one (1) to represent a range of collisions from perfectly inelastic to perfectly elastic collisions. The COR value would usually lie between two ideal types of collisions.

##### 2.2.5.1 Particle-wall collision

The wall is modeled as an unmoving object. Therefore, particle velocity after collision with a wall (Figure 2.9) in the X- and Y-directions can be computed from the definition of the coefficient of restitution (COR):

$$\begin{bmatrix} u_{fx} \\ u_{fy} \end{bmatrix}_p = \text{COR}_{p2w} \begin{bmatrix} u_{0x} \\ u_{0y} \end{bmatrix}_p, \quad (2-69)$$

where  $u_0$  is the velocity before the collision,  $u_f$  is the velocity after collision,  $\text{COR}_{p2w}$  is the COR for collision between particle and wall, and the subscripts X and Y represent the velocity in X- and Y-directions.

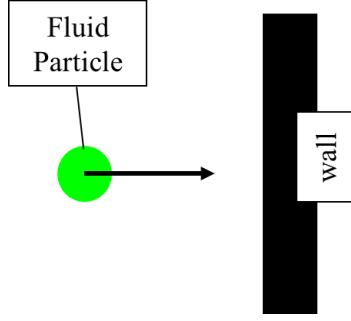


Figure 2.9 Collision between fluid particles and wall

The fluid particles are permitted to flow over wall surfaces to simulate the overflow of runout material over a barrier. The particles flowing over the walls (henceforth, overflowing particles) are assumed to travel vertically or very nearly vertical slopes; therefore, the overflowing particles are airborne and only subjected to the kinetic energy (KE) and gravitational potential energy (GPE). Furthermore, no entrainment is assumed as the wall materials are constructed from a rock or reinforced concrete.

The SPEC-debris method models a boundary condition at the outer edges of the digital elevation model (DEM). The velocity of the particle is reduced by the COR when colliding with the DEM boundary, as defined in Eq. (2-68). However, unlike the walls, the particles cannot pass or flow over the DEM boundary.

### 2.2.5.2 Particle-particle collision

As the particle rebounds and flows around a wall, the particle loses kinetic energy from turbulent flow and fluid diffusion occurs as fluid particles collide randomly. The SPEC-debris method accounts for the effect of velocity change from turbulence and diffusion by simulating the momentum transfer from the collision between fluid particles (Figure 2.10). Assuming no angular momentum transfer occurs from particle-to-particle collisions, the coefficient of restitution (COR) and the conservation of linear momentum are used to compute the particle velocity after collision in the X- and Y-directions:

$$\begin{aligned} \begin{bmatrix} u_{fx} \\ u_{fy} \end{bmatrix}_1 &= \frac{1}{m_1 + m_2} \left( \begin{bmatrix} \cos(\theta_1 - \theta_c) \cdot (m_1 - m_2 \text{COR}_{p2p}) \\ \cos(\theta_2 - \theta_c) \cdot m_2 (1 + \text{COR}_{p2p}) \end{bmatrix} \cdot \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \right) \begin{bmatrix} \cos \theta_c \\ \sin \theta_c \end{bmatrix} + u_1 \sin(\theta_1 - \phi) \begin{bmatrix} -\sin \theta_c \\ \cos \theta_c \end{bmatrix}, \\ \begin{bmatrix} u_{fx} \\ u_{fy} \end{bmatrix}_2 &= \frac{1}{m_1 + m_2} \left( \begin{bmatrix} \cos(\theta_2 - \theta_c) \cdot (m_2 - m_1 \text{COR}_{p2p}) \\ \cos(\theta_1 - \theta_c) \cdot m_1 (1 + \text{COR}_{p2p}) \end{bmatrix} \cdot \begin{bmatrix} u_2 \\ u_1 \end{bmatrix} \right) \begin{bmatrix} \cos \theta_c \\ \sin \theta_c \end{bmatrix} + u_2 \sin(\theta_2 - \phi) \begin{bmatrix} -\sin \theta_c \\ \cos \theta_c \end{bmatrix}, \end{aligned} \quad (2-70)$$

where  $u_0$  is the velocity before the collision,  $u_f$  is the velocity after collision,  $\text{COR}_{p2p}$  is the COR for collision between particles,  $\theta_n$  is the travel angle of the  $n^{\text{th}}$  particle,  $m_n$  is the mass of the  $n^{\text{th}}$  particle,  $\theta_c$  is the contact angle between particles at collision, and the subscripts X and Y represent the velocity in X- and Y-directions.

The mass of the particle is computed by multiplying the density and particle volume.

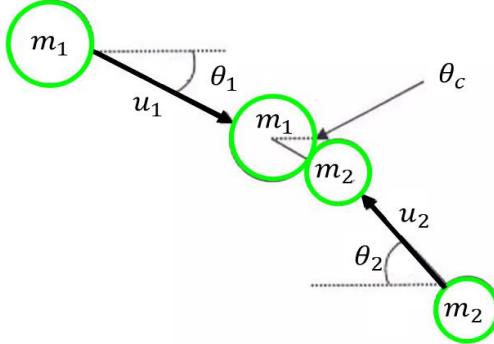


Figure 2.10 Collision between two fluid particles

The SPEC-debris method detects a collision between two particles based on the center-to-center distance between the particles using the following rule:

$$\text{collision detection} = \begin{cases} \text{collision,} & \text{if } s_{ij} \leq (R_1 + R_2) \\ \text{no collision,} & \text{else} \end{cases} \quad (2-71)$$

where  $s_{ij}$  is the distance between particles computed by Eq. (2-11), and  $R$  is the radius of the particle.

More than one collision is expected to occur between the particles. Instead of solving the complex N-body rigid body collision with multiple collisions, the SPEC-debris method assumes that the resultant effect of the multiple collision is the cumulation of multiple one-to-one particle collisions. Therefore, the particle velocity ( $u_f$ ) and travel direction ( $\theta_V$ ) after the collision are solved iteratively by updating the particle  $u_f$  and  $\theta_V$  from each collision with other colliding particles.

## 2.4 Summary

The SPEC-debris runout simulation method has been derived from solving the following principles using the explicit analysis with reasonable assumptions and approximations:

- smoothed particle hydrodynamics (SPH) interpolation,
- pathway algorithm,
- energy conservation model,
- entrainment model, and
- collision mechanics.

The simulation method can be utilized in various analyses, from debris-flow propagation to the performance of closed- and open-type barriers. The verification of the SPEC-debris method is described in Chapter 5.

## Chapter 3. Optimal Barrier Design Framework

### 3.1 Introduction

Debris-flow, a runout hazard caused by rapidly-flowing debris mass, is considered one of the most hazardous disasters that lead to infrastructure damages and casualties (Jakob and Hungr, 2005). The scale and frequency of debris-flow hazards are increasing due to more intense rainfall and long rainstorm duration due to climate changes, which results in casualties and infrastructural damages (Park et al., 2018). Mitigating debris-flow hazards with full evacuation relies on a reliable and accurate early warning system, which is difficult due to high uncertainty in the weather and groundwater conditions. Even with an accurate early warning system, the rapid speed of debris-flow may provide insufficient time for evacuation.

Barriers, such as check dams and slit-wall barriers, are often designed due to their reliability for effective mitigation measures against debris-flow hazards. The function of a closed-type barrier is to retain the volume of debris flow by confining the debris flow. In comparison, the open-type barriers are designed to retain large debris, such as rock boulders and trees, and attenuate the debris-flow speed. In addition, a barrier is built in a valley or channel using various materials from gabions, soil, stone, concrete, or wood (Lucas-Borja et al., 2019).

Designing barriers as effective mitigation against debris-flow hazards requires the engineers to consider the following factors of the barrier:

- location – barrier should be placed on the path of debris-flow propagation, i.e., flowpath,
- type – appropriate barrier type should be selected based on the design requirement,
- performance – barrier performance against debris-flow should ensure successful mitigation,
- size – adequate barrier size is required to avoid overflow of debris-flow and maintain structural integrity at the moment of collision to prevent further damage by initiating a dam break (Rossi and Armanini, 2019), and
- cost – the cradle-to-grave cost of the barrier design should be considered for sustainable design.

These factors for designing barrier mitigation systems are not independent of each other. For example, barriers placed near the toe of the slope would be more accessible during construction and maintenance. However, the debris-flow characteristics (volume, depth, and velocity) would be larger near the toe of the slope; hence, a larger barrier size would be required. For another example, the barrier type should be considered based on the required barrier performance. An open-type barrier cannot completely halt the debris-flow propagation after a collision like a closed-type barrier. Due to the complexity of the barrier design, an iterative engineering design approach is required to converge to an optimal barrier design that satisfies multiple interdependent design factors.

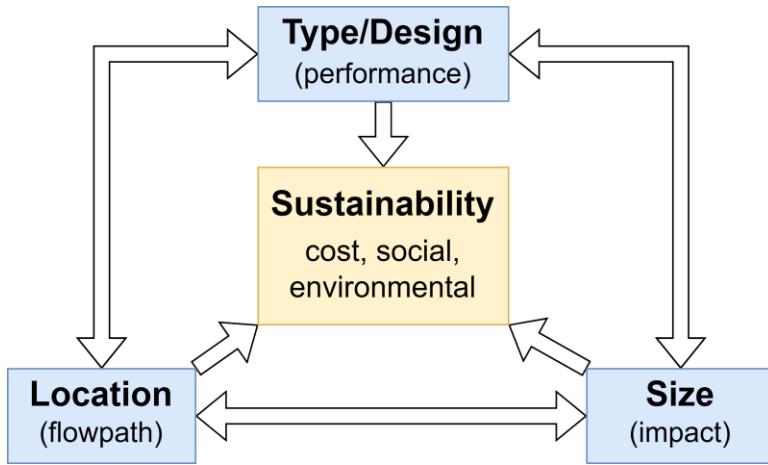


Figure 3.1 Interdependent factors for designing optimal barriers

This study developed an efficient barrier design framework to provide a generalized procedure for selecting the optimal barrier design. The developed framework incorporates the SPEC-debris simulation model from Chapter 2; therefore, the engineer can directly use the debris-flow simulation results from SPEC-debris for an optimal barrier design. The developed framework uses a more efficient iterative search method with an optimization search algorithm to efficiently determine an optimal barrier design satisfying multiple interdependent factors, such as the location, size, and type/design of barriers. The purpose of Chapter 3 is to detail the developed optimal barrier design framework.

### 3.2 Review of Existing Barrier Design Framework

The main approaches used to design barriers are based on the following debris-flow analysis methods: empirical and numerical analysis models.

#### 3.2.1 Empirical analysis approach

The overall design process of barrier with an empirical analysis model utilizes empirical debris-flow models (see Section 2.1) developed from the debris-flow case studies and relies heavily on the engineer's judgment. The barrier design procedure is split into the following steps:

- (1) site selection – determine the landslide region, i.e., source, and the channel at which debris-flow is expected to propagate, i.e., debris-flow flowpath,
- (2) hazard analysis – compute debris-flow propagation along the selected debris-flow flowpath,
- (3) initial design – select barrier location and size based on the hazard analysis result, and
- (4) detail design – analyze barrier performance based on the impact pressure at the barrier wall and debris-flow collision.

Simple guidelines are suggested for selecting barrier locations; for example, the “Riverbank Erosion Control and Mitigation Institution” in South Korea for selecting suitable barrier locations (a) at a narrow valley after up-stream with a wide area, (b) at the debris-flow channel close to the down-stream, and (c) on top of exposed rock surfaces (Korean Design Standard, 2018).

Barrier design with empirical models is a simple and powerful analysis tool. The design procedure is split into several simple stages and simplifies the complex design process; therefore, an engineer can quickly iterate between each stage to converge upon an optimal barrier design.

The limitation of the empirical model described in Section 2.1 is also applicable to the design procedure relying on the empirical model. Determining debris-flow flowpath is complicated or unidentifiable, especially for regions with wide and relatively flat inclinations. The error is transferred along with the design procedure; for example, an inaccurate debris-flow flowpath would result in inaccurate barrier location and size. In addition, the guidelines alone are insufficient to select the optimal barrier locations. Furthermore, every debris-flow-prone region varies widely in topography and geology; therefore, the suggested barrier locations may not be applicable and cannot be guaranteed to provide an optimal barrier location.

The limitations listed above prevent the simple barrier design process based on the empirical models from being a generalized design procedure. Therefore, the developed optimal barrier design framework utilizes a numerical simulation method for analyzing debris-flow hazards and barrier performance.

### 3.2.2 Numerical analysis approach

As described in Section 2.1, several three-dimensional (3D) numerical analysis methods from the depth-averaged continuum models (Hung, 1995; McDougall, 2006; FLO-2D Software Inc., 2007; Hung and McDougall, 2009; Christen et al., 2010; Pastor et al., 2014; Mergili et al., 2017), continuum models (Soga et al., 2016; Jeong and Lee, 2019; Domínguez et al., 2021; Kim et al., 2021), and discontinuum mechanics (Shen et al., 2018; Li and Zhao, 2018) have been utilized to analyze debris-flow runout and barrier performance. Unlike the empirical model, the numerical simulation can be used in general situations and analyze barriers with any geometry. Therefore, debris-flow numerical simulation has been performed by inputting geotechnical, rheological, and geomorphological properties.

The current barrier design procedure utilizes the numerical simulation and follows these general steps to analyze the influence of barriers on debris-flow hazards:

- (1) reference simulation – perform runout simulation without any barrier,
- (2) barrier design – select barrier location and design based on the reference simulation,
- (3) test – analyze the performance of the barrier and change of debris-flow characteristics (velocity, depth, and volume) due to the presence of the barrier, and
- (4) iteration – repeat steps (2) and (3) by changing the barrier design.

Typically, several barrier design scenarios are created, and each design scenario is analyzed with the numerical simulation. Then, the optimal barrier design is selected from the barrier design scenarios based on the barrier performance.

The barrier design approach with the numerical analysis provides a generalized procedure to explicitly predict the debris-flow flowpath and impact (volume, depth, velocity, and pressure) for sites without landslide or debris-flow inventory. Therefore, the numerical simulation has been the standard approach for designing barriers and analyzing their performance.

However, the optimization procedure is considered inefficient. The optimal barrier selection process described previously is a brute-force iterative method. The brute-force method requires the simulation of all potential barrier locations or a sufficiently large number of cases to guarantee the optimality of the barrier design. However, increasing the number of barrier design scenarios would make the currently prevalent optimal barrier location search methods computationally expensive and infeasible to be solved. In order to circumvent the significant disadvantage of the brute-force method, researchers often limit the number of design scenarios; for example, four design scenarios were considered in the Choi and Kwon (2021) paper for selecting the optimal barrier location, as shown in Figure 3.2.

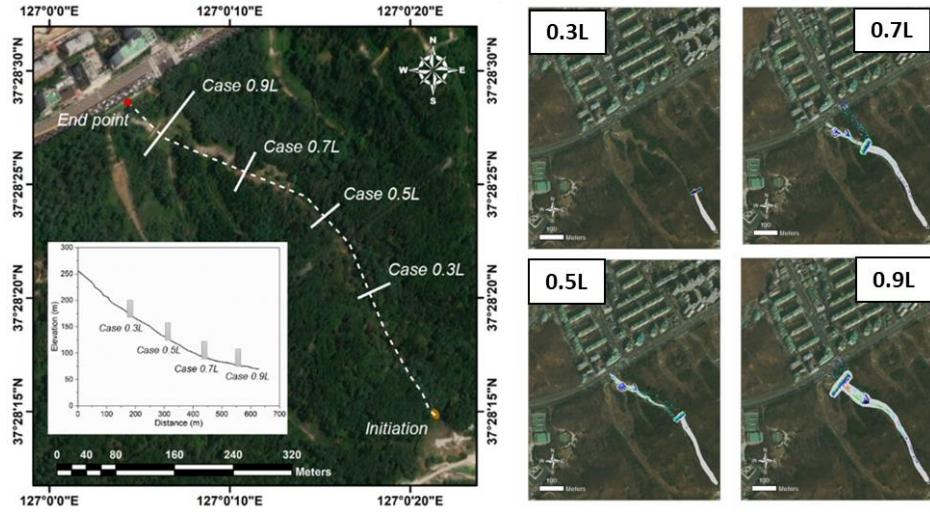


Figure 3.2 Barrier design scenarios and DAN3D debris-flow simulation results (Choi and Kwon, 2021)

Hence, the developed optimal barrier design framework utilizes a more efficient heuristic optimization algorithm to determine suitable barrier design and locations. Finally, the optimization procedure allows computing to quickly search for an optimal barrier location and design by considering multiple design aspects.

### 3.3 Optimal Barrier Design Framework

#### 3.3.1 Framework overview

The optimal barrier design framework adopts the design stages of the barrier design with an empirical analysis approach by separating hazard assessment, initial design, and detailed design into different stages. Although the detailed design stage relies on the traditional numerical analysis method, the hazard assessment and the initial design stages have been modified so that the computer can select the optimal initial design based on design criteria set by the engineer. The optimal barrier design framework procedure follows these steps:

- (1) 3D debris-flow simulation – perform debris-flow hazard assessment through a 3D runout simulation model,

- (2) debris-flow cluster analysis – group debris-flow as clusters with representative debris-flow characteristics (volume, velocity, depth, and impact pressure),
- (3) debris-flow network – construct a data structure for the debris-flow cluster analysis,
- (4) optimal barrier location selection – select the optimal barrier location satisfying multiple optimization criteria, and
- (5) detailed barrier design – analyze the performance of the barrier design through a numerical analysis.

The optimal barrier design framework's main innovative features from other existing methods are stages (1) to (4). The first three stages transform the hazard assessment results into a debris-flow network data structure, which is utilized at stage (4) by the optimization algorithm for selecting the optimal barrier location based on the design parameters and optimization criteria. The flowchart in Figure 3.3 describes each stage and the data transferred between the stages.

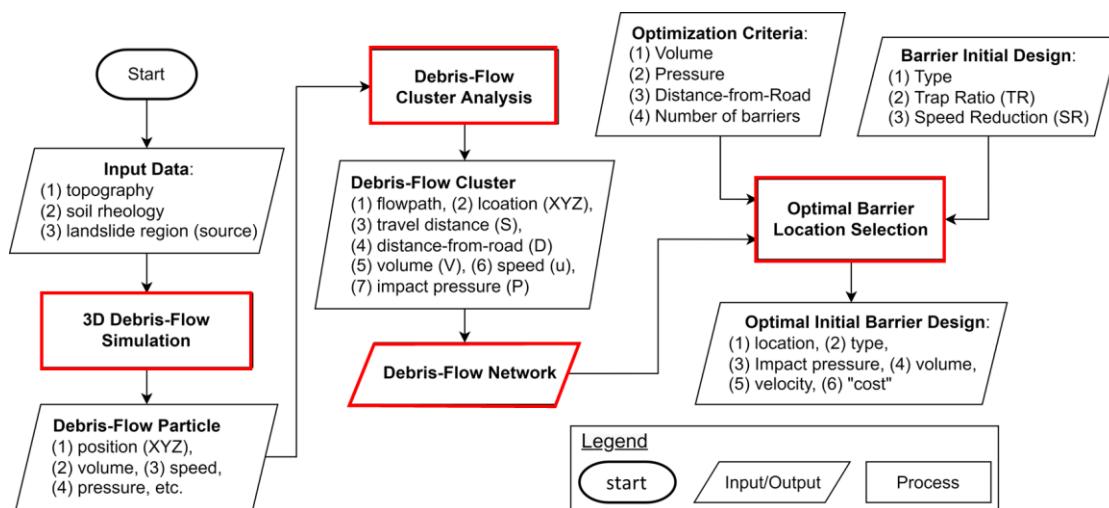


Figure 3.3 Optimal barrier design framework flowchart

### 3.3.2 3D debris-flow simulation

The debris-flow simulation is utilized to perform debris-flow hazard assessment. The debris-flow simulation method should include 3D terrain and entrainment phenomena. The runout simulation model should account for the 3D terrain to identify the flowpath of debris-flow. As barriers should be placed along the flowpath, the flowpath from the simulation signifies potential locations to install barriers.

Another critical component of debris-flow simulation is the entrainment phenomena. For example, the debris-flow erodes the topsoil while flowing and transports the eroded soil mass along with the debris-flow; therefore, debris-flow increases in volume. Entrainment is considered an important phenomenon that should be incorporated into the debris-flow numerical simulation (Hung, 1995; Hung et al., 2005; Iverson, 2012; Lee et al., 2019).

The developed optimal barrier design framework utilizes the SPEC-debris simulation method, as described in Chapter 2. The developed numerical method accounts for the 3D terrain and incorporates a semi-

empirical entrainment model (Section 2.2.4.1). Furthermore, the developed simulation method can analyze barrier performance; hence, it is utilized for the detailed barrier design stage of the optimal barrier design framework. Finally, the rest of the chapter will describe how the SPEC-debris results are utilized.

### 3.3.3 Debris-flow cluster analysis

The SPEC-debris simulation method discretizes the fluid continuum body into particles; therefore, each debris-flow characteristic (volume, depth, and velocity) from the SPEC-debris simulation are given for each debris-flow particle. Tracking individual debris-flow particles is neither efficient nor precise. As each particle is part of the continuum debris-flow body, the fluid particles should be considered as a whole to determine the state of the debris-flow. A group of debris-flow particles that behave as a single continuous mass is defined as a debris-flow cluster, as shown in Figure 3.4. The developed debris-flow cluster analysis combines the properties of debris-flow particles to compute collective properties (volume, velocity, depth, and impact pressure) of debris-flow clusters. The overall flowchart of the framework is shown in Figure 3.5, and the processes are discussed in the following section.

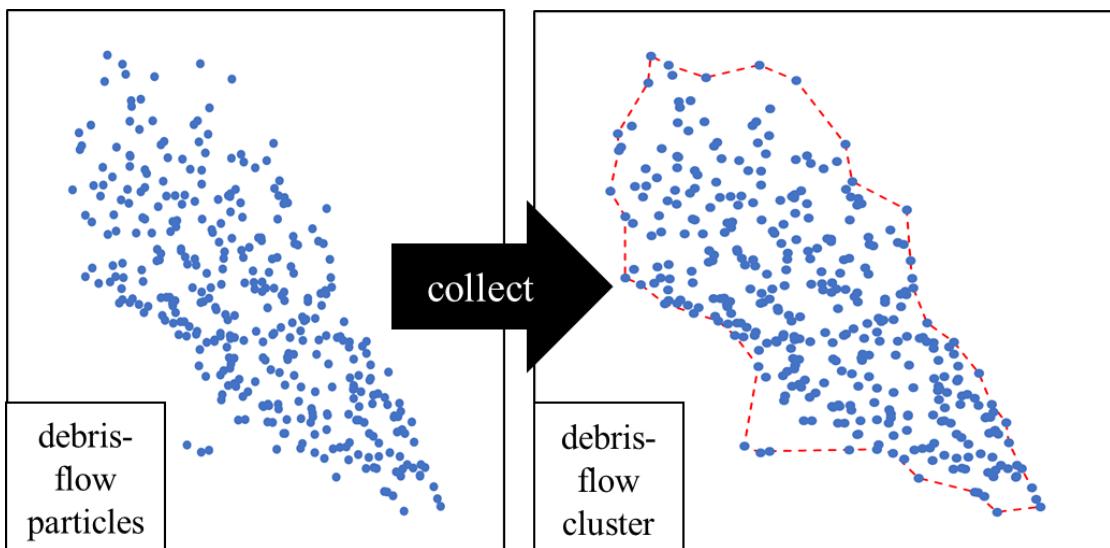


Figure 3.4 Debris-flow cluster as a collection of debris-flow particles

#### 3.3.3.1 Clustering debris-flow particles

The positions of debris-flow particles at every simulation time interval are extracted from the SPEC-debris simulation results. Actual fluid-like debris-flow is a continuous mass; hence, the particles are collected into a cluster to transform the particles into a single entity. Therefore, the k-means++ clustering algorithm (Lloyd, 1982; Arthur and Vassilvitskii, 2007), an unsupervised machine learning algorithm, efficiently sorts the particles into different clusters based on the initial source locations. The algorithm sorts the particle by classifying each particle into different clusters by minimizing the within-cluster sum of squares (WCSS).

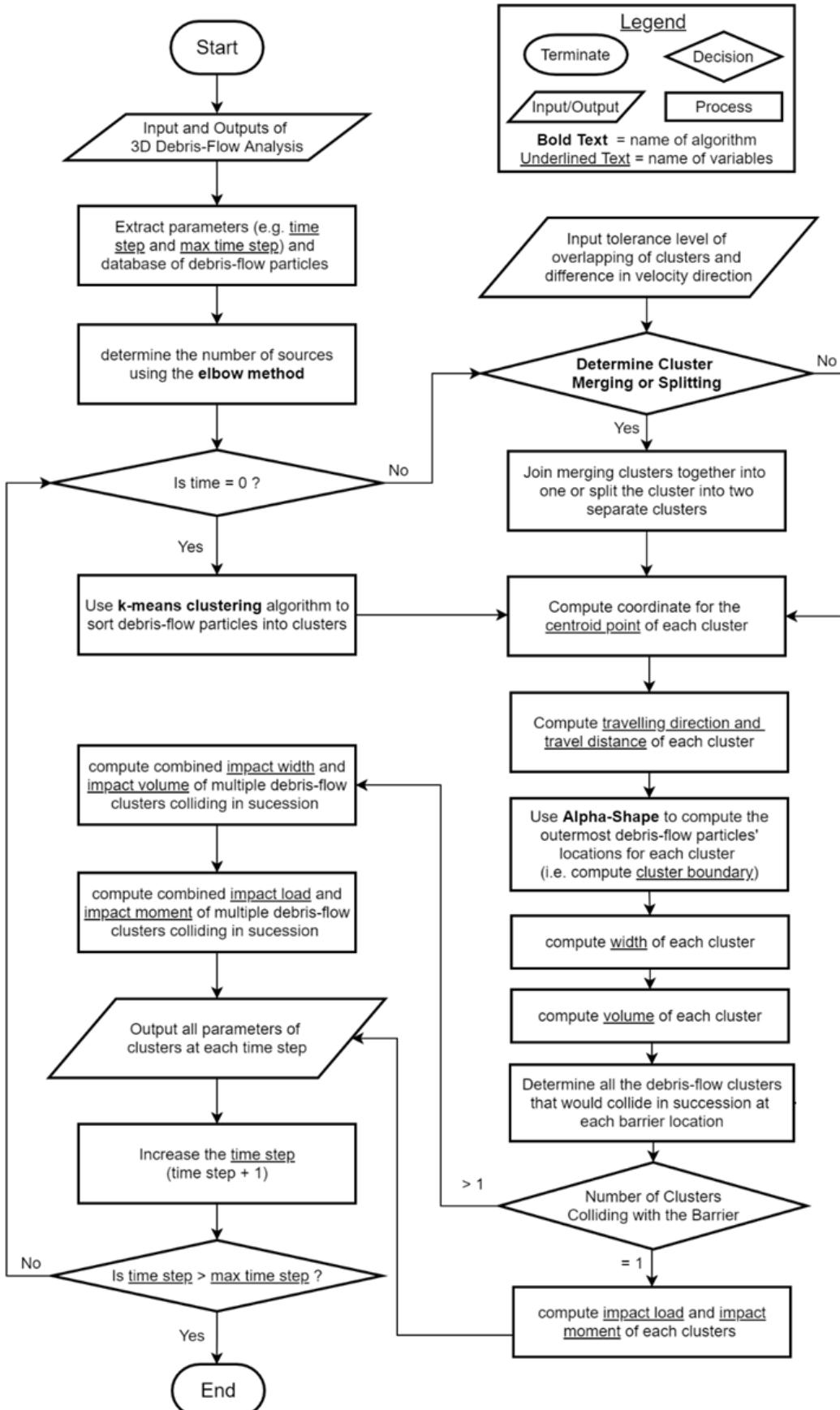


Figure 3.5 Overall flowchart of debris-flow cluster analysis (Cheon et al., 2020)

$$WCSS = \sum_{j=1}^k \sum_{i=1}^n \sqrt{(x_{c,j} - x_{i,j})^2 + (y_{c,j} - y_{i,j})^2}, \quad (3-1)$$

where  $k$  is the total number of clusters;  $n$  is the total number of points classified to cluster  $j$ ;  $x_{c,j}, y_{c,j}$  are the x- and y-coordinates of the centroid point of cluster  $j$ ; and  $x_{i,j}, y_{i,j}$  are the x- and y-coordinates of data points classified to cluster  $j$ .

The value of  $k$  in the k-means clustering algorithm was set to be equal to the number of runout mass source locations. For example, in the debris-flow cluster analysis, the Elbow method (Ketchen and Shook, 1996) was utilized to automatically detect the value of  $k$  for a given layout of initial particle positions.

The k-means clustering algorithm follows these procedures, as shown in Figure 3.6:

- (1) initialization - select  $k$  number of the positions of the particles to be used as the initial centroid point of the clusters,
- (2) partition – partition the particles using the Voronoi diagram (Aurenhammer, 1991),
- (3) new centroid – compute cluster centroid by averaging the particle positions with Eq. (3-2), and
- (4) termination - repeat steps (2) and (3) until WCSS converges to a minimum value.

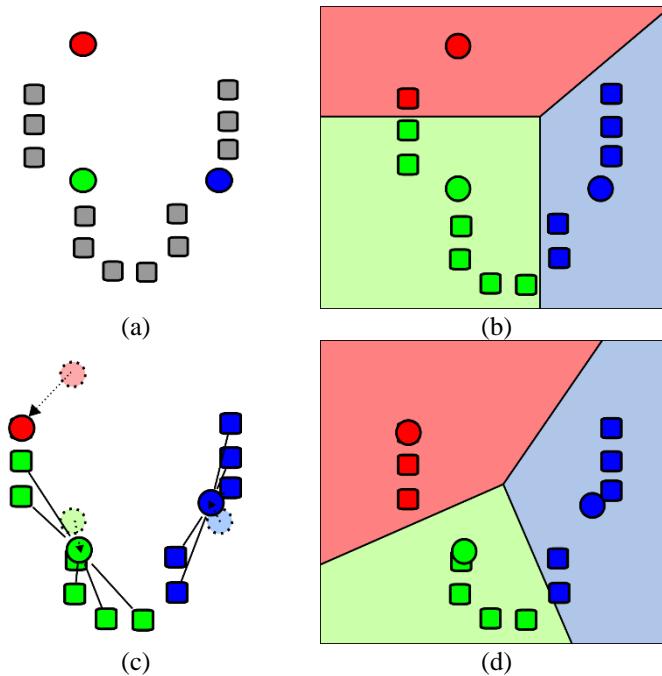


Figure 3.6 A simple demonstration of the k-means clustering algorithm (Weston.pace, 2007):

- (a) initialization, (b) partition, (c) new centroid, and (d) termination.

$$x_{c,j} = \frac{1}{n} \sum_{i=1}^n x_{i,j}; \quad y_{c,j} = \frac{1}{n} \sum_{i=1}^n y_{i,j}, \quad (3-2)$$

where  $n$  is the total number of particles classified to cluster  $j$ ;  $x_{c,j}, y_{c,j}$  are the x- and y-coordinates of the centroid point of cluster  $j$ ; and  $x_{i,j}, y_{i,j}$  are the x- and y-coordinates of data points classified to cluster  $j$ .

Combining the k-means++ clustering algorithm and the Elbow method can find the number of sources automatically, classify the debris-flow particles into debris-flow clusters, and compute the centroid of each cluster for any general cases. This study utilized the k-means clustering function in the scikit-learn Python library (Pedregosa et al., 2011).

### 3.3.3.2 Debris-flow cluster boundary

Each debris-flow cluster is a collection of debris-flow particles; therefore, the boundary of the debris-flow cluster is defined by the positions of the outermost debris-flow particles. The  $\alpha$ -shape (alpha-shape) algorithm (Edelsbrunner et al., 1983) determines the outermost debris-flow particles. The  $\alpha$ -shape is a generalized solution to the Convex Hull problem (de Berg et al., 2000); therefore, the polygon formed by the boundary points computed by the  $\alpha$ -shape algorithm can have a concave shape, unlike the Convex Hull. For example, the Convex Hull algorithm would overestimate the extent of the debris-flow cluster to distribute the debris-flow particles, as shown in Figure 3.7.

The  $\alpha$ -shape algorithm requires a predefined  $\alpha$  (alpha) value to determine the outermost particle locations. A simple iterative search method has been developed to determine the optimal  $\alpha$  (alpha) value for a given distribution of debris-flow particles.

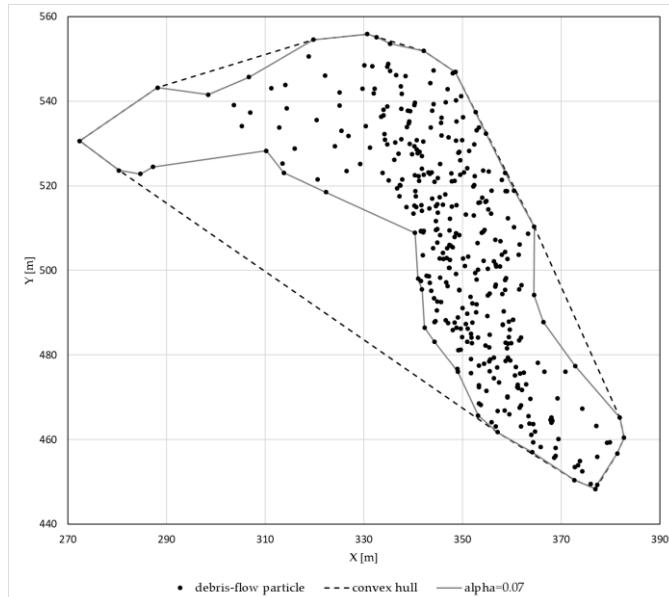


Figure 3.7 Comparison of cluster boundary from the Convex Hull (dotted line) and the  $\alpha$ -shape (solid line) algorithm (Cheon et al., 2020)

For a given layout of debris-flow particles, the optimal value of alpha ( $\alpha$ ) used for the  $\alpha$ -shape algorithm was determined by the following procedure:

- (1) find the boundary points by conducting the  $\alpha$ -shape algorithm with  $\alpha = 0$ , i.e., solve a Convex Hull problem,
- (2) compute the area and the perimeter of the polygon made from the boundary points,

- (3) repeat steps (2) and (3) after incrementing the  $\alpha$  value by a fixed value like 0.01, and
- (4) determine whether the value of  $\alpha$  selected satisfies the following optimal value criteria:
  - The change of the area and the perimeter computed by the current  $\alpha$  value relative to the area and the perimeter computed by the previous iteration of the  $\alpha$  value are below a tolerance level ( $\varepsilon$ ):

$$\begin{aligned} (\text{area}_{\alpha(\text{current})} - \text{area}_{\alpha(\text{previous})}) &< \varepsilon_{\text{area}}, \\ (\text{perimeter}_{\alpha(\text{current})} - \text{perimeter}_{\alpha(\text{previous})}) &< \varepsilon_{\text{perimeter}}, \end{aligned} \quad (3-3)$$

- The boundary polygon generated from the  $\alpha$  value encompasses all debris-flow particles, except those considered outliers. The outlier particles are defined as particles located 1.5 times greater than the average distance between the other points beside the cluster, as shown in Figure 3.8.
- The number of boundary polygons generated from the boundary points by the current  $\alpha$  value does not exceed one (1).

Typically, the optimal  $\alpha$  value did not exceed 0.5 and would not fall below zero (0); hence, the optimal  $\alpha$  value is the smallest value between 0 and 0.5 that satisfies these criteria.

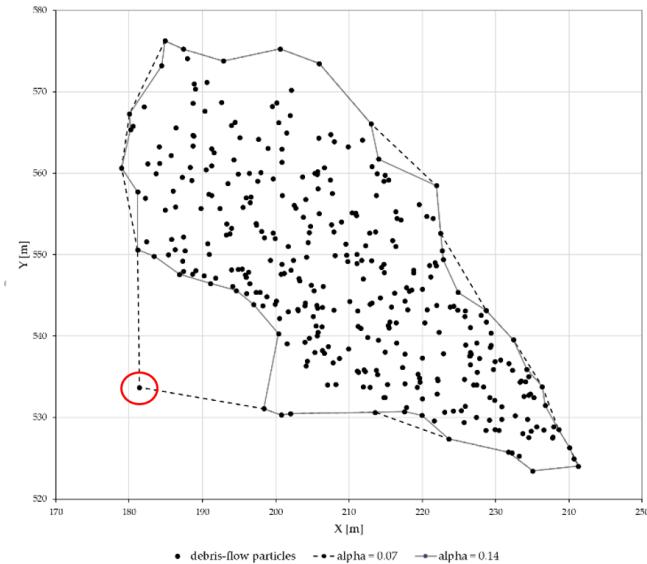


Figure 3.8 Exclusion of outlier point (red circle) in cluster boundary through the selection of alpha value  
(Cheon et al., 2020)

When the boundary of the debris-flow cluster is defined through the  $\alpha$ -shape algorithm, the area and width of the cluster can be determined by the outermost debris-flow particle positions. The area is computed using the shoelace formula with the XY coordinates of the outermost particles. The effective width of the debris-flow cluster is defined as the maximum width of the debris-flow cluster perpendicular to the direction of the debris travel direction, as shown in Figure 3.9.

$$c = y - x \tan \theta, \quad (3-4)$$

$$w = \begin{cases} |\max(\{c\}) - \min(\{c\})| \cos \theta, & |\theta| \neq 90^\circ, \\ |\max(\{x\}) - \min(\{x\})|, & |\theta| = 90^\circ, \end{cases} \quad (3-5)$$

where  $x$  and  $y$  are the X- and Y-coordinates of the boundary points,  $\theta$  is the angle of the travel direction of the debris-flow cluster,  $\{x\}$  is the set containing the x-coordinates of the boundary points, and  $\{c\}$  is the set containing the y-intercept values calculated.

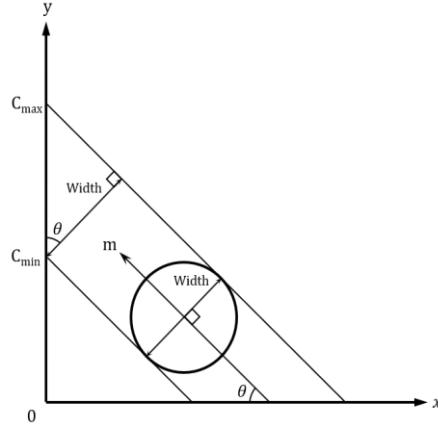


Figure 3.9 The definition of the width of a debris-flow cluster (Cheon et al., 2019)

### 3.3.3.3 Debris-flow cluster representative quantities

Each debris-flow cluster is a collection of debris-flow particles; therefore, we can assume that the behavior of the entire debris-flow cluster is the cumulative behavior of individual debris-flow particles. Accordingly, the behavior of a debris-flow cluster can be assumed to act on the centroid, which can be used to track the location and the characteristics (volume, velocity, depth, pressure) of a debris-flow cluster. This simplification provides computational benefits by reducing the computational load while maintaining relevant information to provide accurate results.

The centroid location, which is defined by Eq. (3-2) from the centroid location of the debris-flow cluster, is used to track the following location-based quantities defined in Figure 3.10: the travel distance ( $S$ ) and the distance-from-road ( $D$ ). The travel distance ( $S$ ) is the cumulative distance the debris-flow cluster traveled from the initial landslide failure location, i.e., the source. The distance-from-road ( $D$ ) is the shortest distance between the debris-flow cluster and the nearest road or transportation route. Assuming that the road is a linear section, the distance-from-road ( $D$ ) is computed by the following equations:

$$D = \frac{(y_{d2} - y_{d1})x_p - (x_{d2} - x_{d1})y_p + x_{d2}y_{d1} - x_{d1}y_{d2}}{\sqrt{(x_{d1} - x_{d2})^2 + (y_{d1} - y_{d2})^2}}, \quad (3-6)$$

where  $(x_{d1}, y_{d1})$  and  $(x_{d2}, y_{d2})$  are the XY-coordinates of two points along the road, and  $(x_p, y_p)$  is the XY-coordinate of the debris-flow cluster.

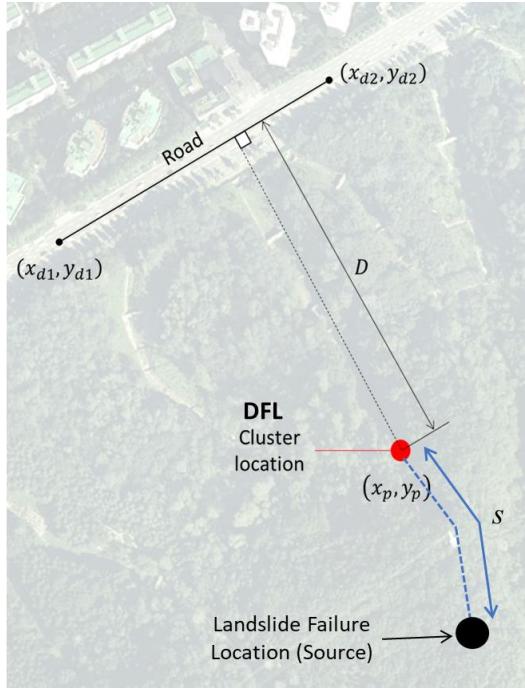


Figure 3.10 The definition of the travel distance (S) and distance-from-road (D)

The volume ( $V_c$ ), velocity ( $u_c$ ), and depth ( $h_c$ ) of the debris-flow cluster quantities are computed by the debris-flow particle quantities:

$$V_c = \sum_{i=1}^n V_i, \quad (3-7)$$

$$u_c = \max[u_1, u_2, \dots, u_i], \quad (3-8)$$

$$h_c = \max[h_1, h_2, \dots, h_i], \quad (3-9)$$

where  $n$  is the number of particles within the cluster, and  $V_i, u_i, h_i$  are the debris-flow particle volume, velocity, and depth, respectively.

The maximum particle value in the debris-flow cluster is selected as the representative debris-flow cluster characteristics. In general, the front velocity and the front depth of the debris-flow body are tracked, which requires determining the front-most particle based on the cluster movement direction and boundary geometry. Usually, the front velocity and front depth match the maximum depth and velocity of the debris-flow propagation; therefore, the maximum particle velocity and depth are selected to represent the debris-flow cluster front velocity and depth.

The Froude number ( $F_r$ ) is a dimensionless ratio between the gravity and inertia to describe different open-channel flows, which is defined as:

$$F_r = \frac{u}{\sqrt{gh}}, \quad (3-10)$$

where  $u$  is the velocity,  $h$  is the depth, and  $g$  is the gravitational acceleration.

The Froude number has been utilized in a debris-flow analysis to overcome the issue of scale. The dimensions of the actual debris-flow hazards are often larger than the debris-flow in laboratory tests. Using the dimensionless factor allows the results from the numerical simulation and laboratory tests to be applicable for computing the impact pressure ( $P_c$ ) of the debris-flow (Costa, 1984; Hübl and Holzinger, 2003; Tiberghien et al., 2007; Scheidl et al., 2013; and Cui et al., 2015). Using the impact pressure empirical model by Cui et al. (2015), the dynamic impact pressure of the debris-flow cluster is defined by the Froude number ( $Fr$ ). The debris-flow cluster is assumed to be a hydrostatic condition when the cluster velocity is zero (0). The pressure of the cluster is defined as:

$$P_c = \begin{cases} 5.3Fr^{-1.5}\rho u_c^2 & , u_c > 0 \\ \rho gh_c & , u_c = 0 \end{cases}, \quad (3-11)$$

where  $P_c$  is the debris-flow cluster pressure,  $Fr$  is the Froude number,  $\rho$  is the density,  $u_c$  is the debris-flow cluster velocity,  $h_c$  is the debris-flow cluster depth, and  $g$  is the gravitational acceleration.

### 3.3.3.4 Debris-flow cluster merge and separation

Multiple debris-flow clusters would flow through the same channel; therefore, the clusters can interact with each other by merging. Additionally, the debris-flow cluster may split into smaller clusters due to the splitting of the flow channel. The debris-flow cluster analysis utilizes simple criteria to determine clusters merging and a cluster splitting.

For merging, both criteria need to be satisfied to consider two debris-flow clusters as merging, as shown in Figure 3.11:

- travel direction - the difference in the travel directions of the debris-flow clusters is within a tolerance limit of  $10^\circ$ , and
- overlapping ratio - the ratio between the number of debris-flow particles located inside the boundary of another cluster over the total number of debris-flow particles in the reference cluster needs to be equal to or larger than the critical overlap area; therefore,

$$\frac{\text{Area}_{(A \cap B)}}{\text{Area}_A} \geq \text{Area Ratio}_{\text{crit}} \text{ and } \frac{\text{Area}_{(A \cap B)}}{\text{Area}_B} \geq \text{Area Ratio}_{\text{crit}}, \quad (3-12)$$

where  $\text{Area}_{(A \cap B)}$  is the overlapping area of debris-flow clusters A and B; and  $\text{Area}_A$ ,  $\text{Area}_B$  are the areas of debris-flow clusters A and B, respectively. The critical value of the overlapping area ( $\text{Area Ratio}_{\text{crit}}$ ) is selected as 0.5 (Cheon et al., 2020).

The depth and velocity of the merged clusters are computed from the average depth and velocity of the combined debris-flow particles. The newly merged cluster combines the volume of each cluster:

$$V_{\text{merged}} = V_A + V_B, \quad (3-13)$$

where  $V_{\text{merged}}$  is the combined volume of merged debris-flow clusters A and B; and  $V_A, V_B$  are the volume of debris-flow clusters A and B, respectively.

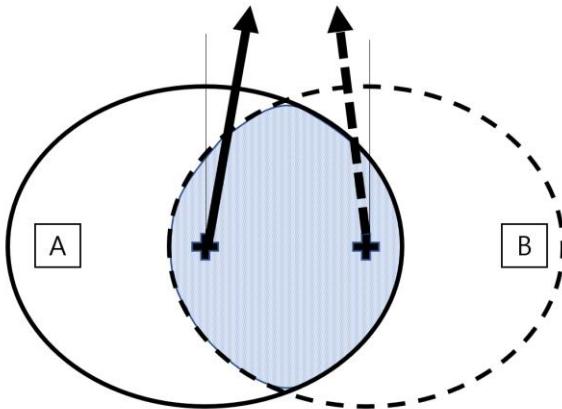


Figure 3.11 Criteria to determine the merging of two debris-flow clusters (Cheon et al., 2020)

For splitting of a debris-flow cluster, the following criteria need to be satisfied:

- centroid position - the average location of debris-flow particles lies outside the boundary of the debris-flow cluster, as shown in Figure 3.12(a), and
- travel angle peaks - a frequency distribution plot displaying the travel direction of debris-flow particles in a cluster shows more than one distinctive peak, as shown in Figure 3.12(b).

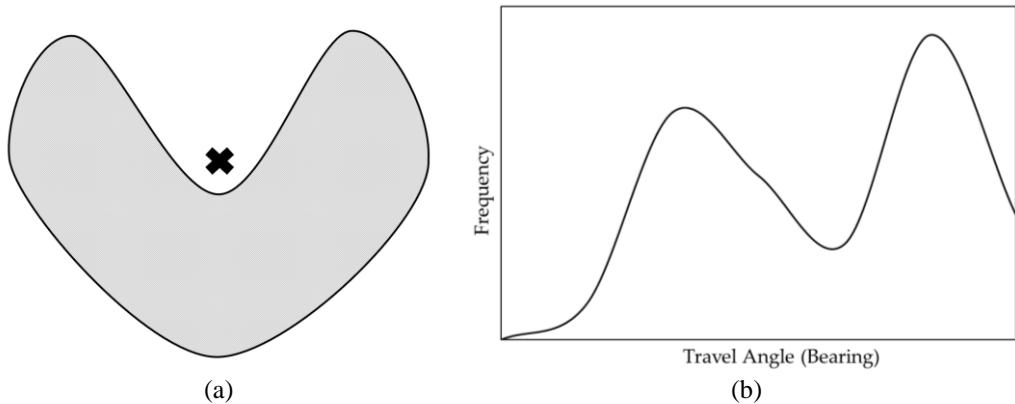


Figure 3.12 Criteria to determine the splitting of a debris-flow cluster (Cheon et al., 2020)

When a debris-flow cluster is determined to be splitting, the debris-flow particles are reclassified using the k-mean++ algorithm into smaller debris-flow clusters. The volume, depth, and velocity of the smaller debris-flow clusters are computed with Eq. (3-7), (3-8), and (3-9).

### 3.3.3.5 Outputs of debris-flow cluster analysis

As shown in the flowchart of the optimal barrier design framework (see Figure 3.3), the debris-flow cluster analysis combines the debris-flow particles into clusters to compute the following representative debris-flow characteristics at each debris-flow cluster location (DFL) along the flowpath:

- location (XYZ-coordinates),
- travel distance (S),
- distance-from-road (D),

- velocity ( $u_c$ ),
- depth ( $h_c$ ),
- volume ( $V_c$ ), and
- pressure ( $P_c$ ).

### 3.3.4 Debris-flow network

The debris-flow cluster analysis combines the debris-flow particles into debris-flow clusters. Each debris-flow cluster location (DFL) quantifies the locations and the characteristics of debris-flow, such as volume, depth, velocity, and pressure; hence, the debris-flow cluster analysis provides location-based debris-flow quantities. However, the time-based debris-flow quantities, such as the debris-flow cluster location at a particular time step, are required to determine the interaction between the clusters. Additionally, the debris-flow flowpath, defined as the path traced as the debris-flow cluster propagates, would provide critical information regarding the suitable locations for barriers. Therefore, in this study, a network-based data structure was developed to store the following debris-flow hazard assessment efficiently: (1) locations of multiple debris-flow flowpaths, (2) intersection between flowpaths, and (3) characteristics of debris-flow along each flowpath.

#### 3.3.4.1 Network structure

The network data structure is a tree-like structure used in graph theory. The network data structure consists of the following components: node and edge. The node represents an object, and the edges represent the object's connection. The connections can be either unidirectional or bidirectional. In the network data structure, the unidirectional edge represents the flow of quantifying from one object to another.

In the optimal barrier design framework, as shown in Figure 3.13, the debris-flow network data structure was utilized to represent the debris-flow cluster location (DFL) and the flow of debris between the DFLs. The nodes represent the DFL objects that contain the debris-flow quantities (volume, depth, velocity, and pressure). The unidirectional edges represent the flow of debris-flow, i.e., flowpath. The flowpath starts from the landslide location, i.e., the source, to the boundary between the urban and mountain regions, henceforth mentioned as the terminus. Therefore, the network structure forms chains of unidirectional links starting from the source and ending at the terminus.

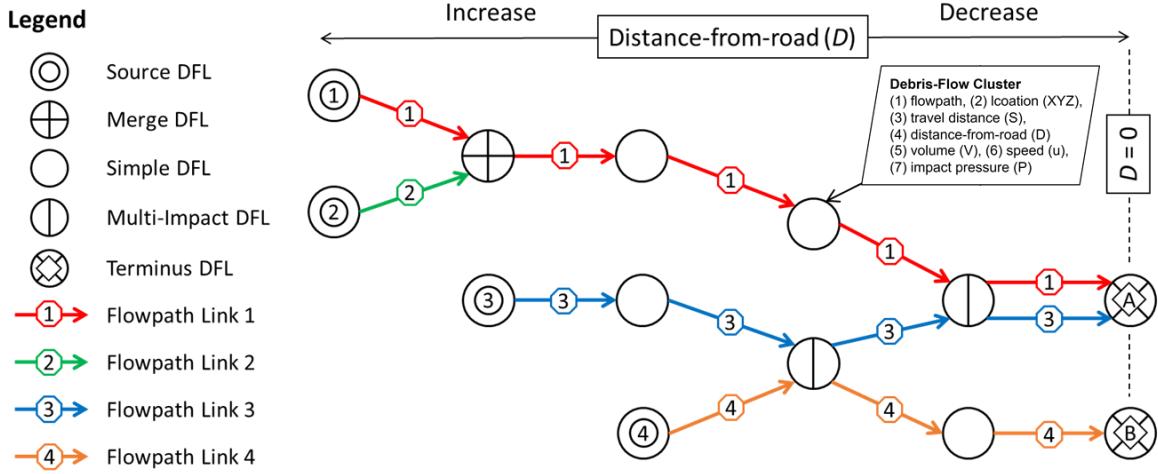


Figure 3.13 Simplified debris-flow network diagram

### 3.3.4.2 Node type

The node types of the debris-flow network data structure provide information regarding the interaction between the different clusters. The debris-flow network categorizes each DFL into one of the following node types:

- source - debris-flow initiation location,
- terminus - at the boundary between urban and mountain regions,
- merge - multiple debris-flow clusters merge and combine into a single larger cluster,
- multi-impact - multiple debris-flow clusters flow through, but no merging occurs, and
- simple – flowpath location not meeting any of the other categories.

The debris-flow network distinguishes the merge and multi-impact based on the temporal interaction between the debris-flow clusters. Two clusters must collide at the DFL simultaneously to be considered a merge node type, as shown in Figure 3.14.

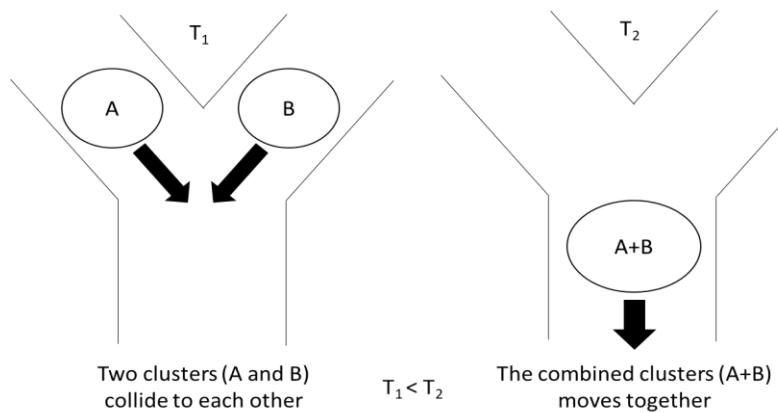


Figure 3.14 Debris-flow cluster location (DFL) with merge node type

By contrast, debris-flow clusters do not simultaneously meet at the multi-impact DFL. Instead, multiple debris-flow clusters flow through the multi-impact DFL without interaction with each other. However, if a barrier is placed at the multi-impact DFL, the barrier would halt the propagation of a debris-flow cluster. Then, other clusters flowing behind the collided cluster will collide against the barrier and merge with the halted debris-flow cluster, as shown in Figure 3.15.

The combined volume loading onto the barrier is computed by the total volume of all debris-flow clusters passing through the multi-impact DFL at the multi-impact DFL. Furthermore, the cumulative impact pressure exerted on the barrier from multiple debris-flow collisions is the cluster impact dynamic pressure summation. The cumulative impact pressure assumes that the impact pressures can be linearly summed, i.e., superposition. Although the actual impact pressure from multiple collisions would not equal the cumulative impact pressure, the current method is simpler and would always compute a conservative result.

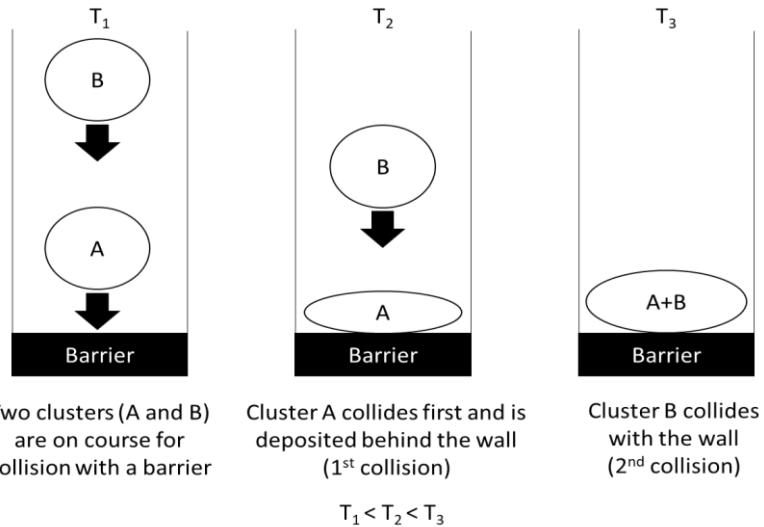


Figure 3.15 Debris-flow cluster location (DFL) with multi-impact node type

### 3.3.4.3 Closed-type barrier mitigation check

In the developed debris-flow network, a series of connected DFLs indicates a flowpath. The ordering of connected DFLs shows the chronological flow of a debris-flow cluster from the source to the terminus. If only closed-type barriers are installed, successful debris-flow mitigation through closed-type barriers is defined as when a set of barriers placed in the flowpaths prevents all the debris-flow clusters from flowing from the source to the terminus. If no debris-flow cluster travels beyond the terminus where the people and the infrastructure are located, debris-flow damage has been successfully mitigated.

Removing a node would also break all the edges connected to the node in a network data structure. Hence, removing a DFL node in the developed debris-flow network would signify the connection between the DFLs, i.e., a flow of debris-flow between the DFLs is broken. A series of connected DFL nodes indicates a flowpath; therefore, removing a DFL node is considered to have broken the flowpath. The debris-flow cluster can no longer flow beyond the location of the removed DFL node.

A closed-type barrier aims to retain the debris-flow volume and prevent flow beyond the wall. Therefore, installing a barrier at the DFL is equivalent to removing a DFL node in the debris-flow network data structure. Hence, the effect of closed-type barriers on the propagation of debris-flow clusters can be simulated using the developed debris-flow network to perform a quick check on mitigation success. For example, suppose a set of closed-type barrier locations severs all connections from the source DFL to the terminus DFL. In that case, the locations of the proposed closed-type barriers are considered to have successfully mitigated the debris-flow disaster, as shown in Figure 3.16.

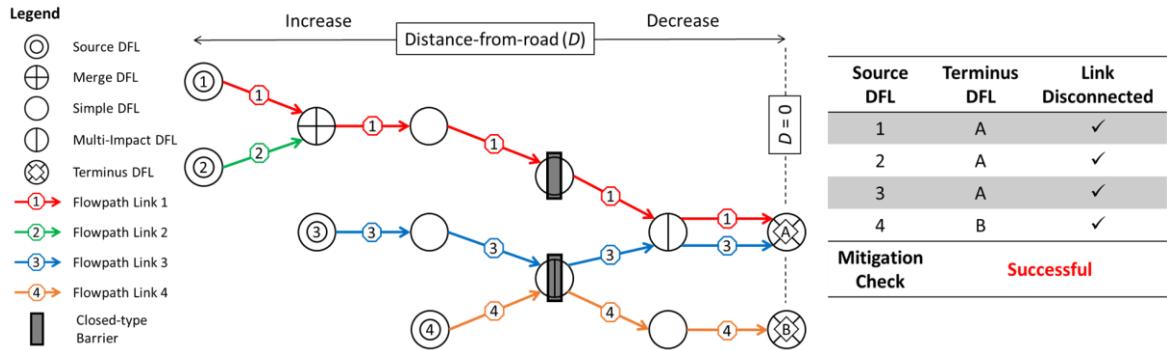


Figure 3.16 Example of successful mitigation for closed-type barrier with debris-flow network data structure

### 3.3.4.4 Opened-type barrier mitigation check

The debris-flow network data structure can also check whether a set of open-type barriers have successfully reduced the debris-flow risk to an acceptable level. Unlike the closed-type barriers, the purpose of an open-type barrier is to reduce the energy of the debris-flow and retain some debris-flow volume. Therefore, the speed ratio (SR) and trap ratio (TR) values represent the effect of the open-type barrier on the debris-flow, i.e., barrier performance.

$$SR = \frac{\text{speed after wall collision}}{\text{speed before wall collision}} = \frac{u_f}{u_0}, \quad (3-14)$$

$$TR = \frac{\text{volume after wall collision}}{\text{volume before wall collision}} = \frac{V_f}{V_0}, \quad (3-15)$$

where  $u$  is the velocity and  $V$  is the volume.

An ideal closed-type barrier would retain all incoming debris-flow without overflowing; hence, the both SR and TR values of an ideal closed-type barrier would be zero (0). For an open-type barrier, the value of SR and TR would lie between zero (0) and one (1), which means the debris-flow can continue flowing after the collision but is reduced in volume and speed.

The reduced and attenuated debris-flow cluster is expected to continue flowing until the cluster reaches the terminus. The terminus is the boundary between the mountain and a developed region, such as an urban or infrastructure. The debris-flow hazard risk is assessed by the potential damage the debris-flow inflicts on a building; therefore, the risk of debris-flow hazard is determined at the terminus location. For example, suppose the installed open-type barriers reduce the debris-flow hazard risk at the terminus location to an

acceptable level. In that case, the open-type barriers can be considered to have successfully mitigated the debris-flow disaster. Therefore, checking the mitigation successfulness of open-type barriers requires computing the debris-flow characteristics, such as volume and speed, at the terminus after barrier collision.

The optimal barrier selection framework utilizes the entrainment model and the energy conservation governing equation of the SPEC-debris runout simulation method for computing the propagation of the debris-flow cluster after colliding with the open-type barrier. The volume increased due to entrainment phenomena is computed with Eq. (2-50) to (2-52) in Section 2.2.4.1. The velocity of the debris-flow cluster is computed with the simplified energy conservation model considering the kinetic energy ( $KE$ ), gravitational potential energy ( $GPE$ ), and the work done from basal friction without the turbulence resistance ( $W_{fr}$ ):

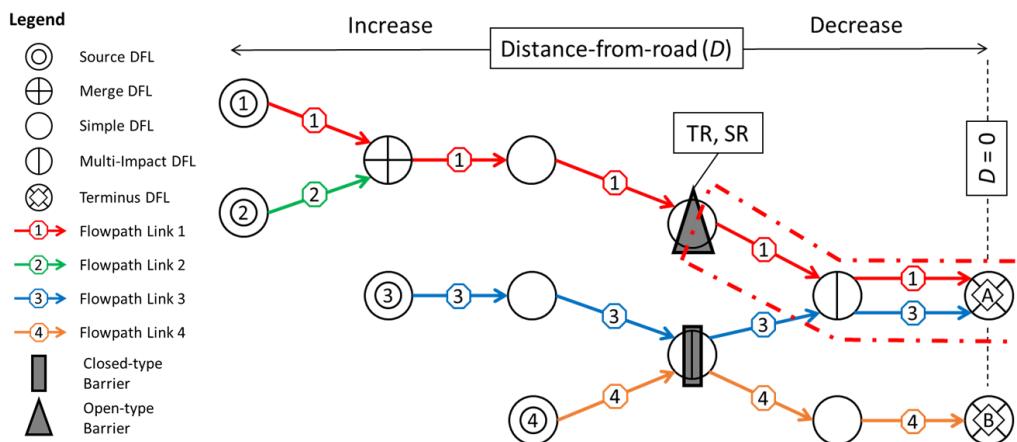
$$KE_f = KE_0 - \Delta GPE - \Delta W_{fr},$$

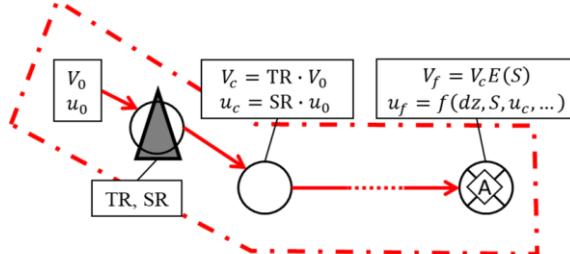
$$E(dS)u_f^2 = u_0^2 - g[1 + E(dS)] [Z_f - Z_0] - 2f_b g \int_0^{ds_i} E(dS) dS, \quad (3-16)$$

where  $E(dS)$  is the entrainment model function,  $dS$  is the traveled distance,  $u$  is the velocity,  $g$  is the gravitational acceleration,  $Z$  is the elevation,  $f_b$  is the base friction coefficient, and the subscripts  $0$  and  $f$  refer to the initial and final states, respectively.

The risk of damage caused by debris-flow hazards is quantified by the vulnerability index ( $VI$ ). Kang and Kim (2017) provide the equation for computing the vulnerability index ( $VI$ ) based on the debris-flow velocity. Furthermore, Kang and Kim (2017) account for the following building structures for computing the  $VI$  value: reinforced concrete (RC) and non-RC structures.

The debris-flow network data structure provides the flowpath information, such as travel distance and elevation change, and the debris-flow characteristics (volume and velocity). Assuming that the debris-flow follows the same flowpath after colliding with the open-type barrier, the developed network structure can be used to compute the propagation of the debris-flow cluster from the DFL with an open-type barrier to the terminus DFLs. Therefore, the mitigation successfulness of the open-type barrier locations can be determined using the debris-flow network data structure.





$\text{TR} = \text{Trap Ratio}$  (volume retention)  
 $\text{SR} = \text{Speed Reduction}$  (speed attenuation)  
 $E(S)$  = entrainment model  
 $f(dz, S, u_c, \dots)$  = energy-based debris-flow model

Figure 3.17 Example of mitigation check for open-type barrier with debris-flow network data structure

### 3.3.5 Optimal barrier location selection

The developed optimal barrier design framework utilizes the stochastic heuristic optimization algorithm to efficiently select the optimal closed and open-type barrier locations satisfying multiple design criteria. The overall procedure for selecting the optimal barrier location is shown in Figure 3.18.

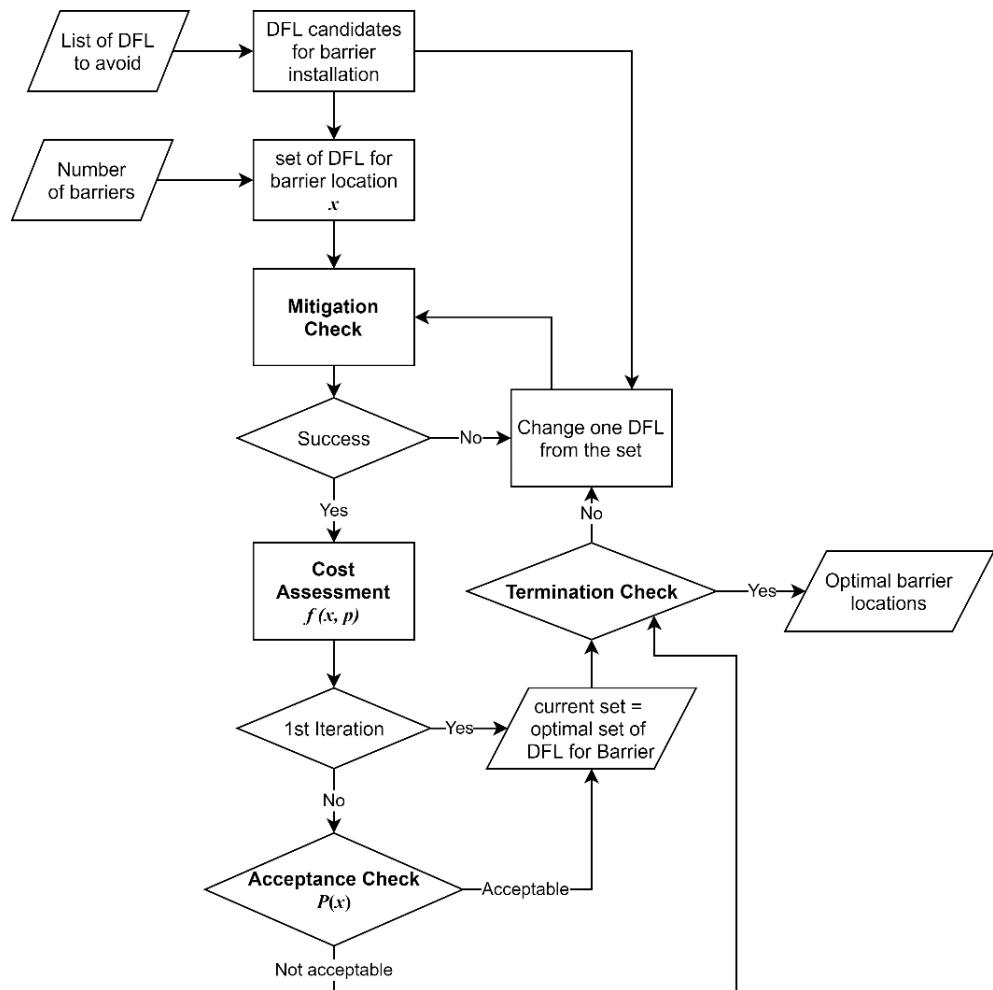


Figure 3.18 Overall flowchart of optimal barrier location selection

### 3.3.5.1 Optimization procedure

Optimization algorithms are utilized to search for a set of barrier locations ( $x$ ) from some set of available choices that yields the minimum cost based on predefined criteria ( $p$ ). The cost (\$) value is computed using a cost function  $f(x, p)$ . The optimization algorithms often use rigorous or heuristic-based iterative approaches to find the optimal solution. Both approaches modify the initial values until an optimal set of barrier locations is found; however, rigorous iterative methods, such as the brute-force method, require computing all possible elements to determine the optimal solution. As the number of barriers and the scale of the debris-flow hazard increase, the brute-force method would be efficient or infeasible to solve within a reasonable time. Hence, a heuristic search method was selected to find the optimal barrier locations.

The heuristic approach solves the optimization search problem by trading speed with optimality, completeness, accuracy, or precision. The heuristic method does not test every possible set of solutions; therefore, the heuristic search method cannot guarantee that the computed solution is the most optimal. There is a possibility that a local minimum solution is computed instead of a global maximum or minimum, as shown in Figure 3.19.

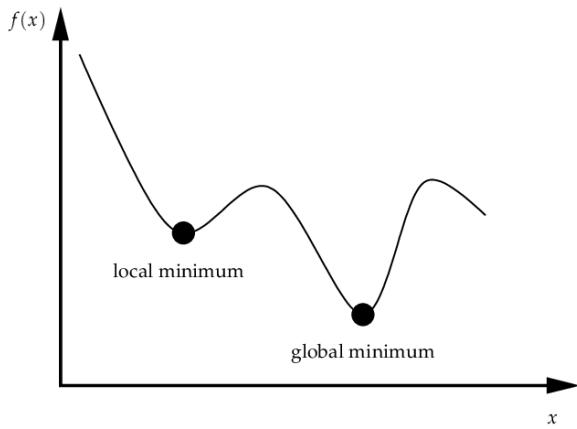


Figure 3.19 Local and global minimum solution

Stochastic heuristic search randomly selects the set of barrier locations that may overcome the issues of being stuck at a local minimum. Furthermore, repeating the entire stochastic heuristic optimization search several times increases the likelihood of the computed set being the global minimum solution and provides greater confidence in the selected optimal barrier locations.

However, the randomly selected barrier locations results may repeat a previously-evaluated barrier set and would be inefficient. Therefore, the optimal barrier location selection method utilized the stochastic tabu search algorithm. The tabu in the selected optimization algorithm excludes the previously evaluated set of barrier locations. As shown in Figure 3.20, the procedure of the stochastic tabu search algorithm follows these steps to find the set of barrier locations yielding the smallest “cost” (\$):

- (1) create an initial optimal set of debris-flow locations (DFLs) as the barrier locations,
- (2) while avoiding a previously checked set of barrier locations, modify the optimal DFL set by randomly replacing one of the DFLs with another,

- (3) compute the “cost” (\$) of the set of barrier locations from DFL data,
- (4) replace or keep the set of optimal barrier locations by comparing “cost” (\$), and
- (5) repeat steps (2) to (4) until the number of iterations has exceeded the maximum iteration limit.

### 3.3.5.2 Optimization criteria and cost analysis

The “cost” (\$) is a dimensionless quantity that objectively quantifies the optimality of the barrier locations. For optimal barrier location selection, the “cost” is evaluated based on the following optimization criteria: mitigation successfulness, number of barriers (N), volume (V), impact pressure (P), and distance-from-road (D).

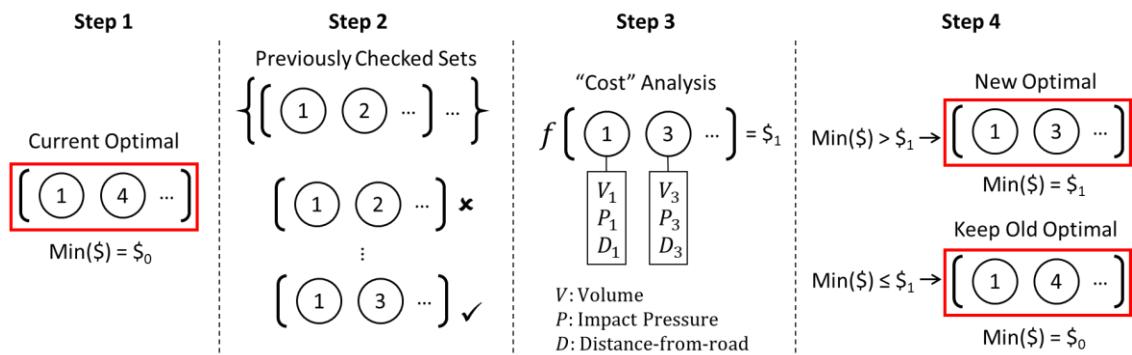


Figure 3.20 Procedure of stochastic tabu search optimization algorithm for optimal barrier location selection

The barrier design aims to successfully mitigate the debris-flow disaster and prevent casualties and infrastructure damages. If a selected set of barrier locations does not succeed in its objective, the barrier locations are unacceptable. In the optimal barrier location selection method, the “cost” of barrier locations failing the mitigation against debris-flow is considered an infinite cost.

The number of barriers (N) provides an indirect measure of environmental impact and the overall monetary cost of the barrier. Installation of barriers usually entails modification of the mountain terrain and the forest. Therefore, achieving successful mitigation against debris-flow disasters with the lowest barriers would minimize the environmental impact, one of the critical evaluation criteria for sustainable development.

When both opened-type and closed-type barriers are considered for optimal barrier design, the open-type barrier number ( $N_O$ ) and the closed-type barrier number ( $N_C$ ) are separately considered. The closed-type barrier requires removing deposited debris behind the barrier after a debris-flow disaster to ensure barrier performance. However, removing deposited debris behind an open-type barrier is more manageable, and barriers can be easily reused after a disaster. Therefore, the maintenance cost of an open-type barrier is considered to be more economical than a closed-type barrier.

The volume (V) and impact pressure (P) of debris-flow indirectly represent the material cost of the barriers. A sufficiently tall barrier is required to ensure the barrier has sufficient volume capacity and prevent excessive overflow over the wall. Furthermore, the higher impact load imposed by the debris-flow would result in thicker walls to ensure structural stability. Therefore, the optimal barrier locations selection method would search for debris-flow locations (DFLs) that minimize the volume (V) and impact pressure (P).

The distance-from-road (D), as defined in Section 3.3.3.3, represents the accessibility of the barrier locations. Accessibility of a site changes the construction and maintenance costs. Higher accessibility implies easier vehicle access between the barrier location and a road. Vehicle access would enable more accessible transportation of construction material and machinery. Furthermore, a larger digger can be brought to the site to remove deposited debris, wood, and boulders behind the barrier. Therefore, DFL with the lowest distance-from-road (D) value implies more economical construction and maintenance costs.

In the optimization algorithm procedure stage (3), i.e., cost analysis, the “cost” (\$) of the set of DFLs is computed by considering these five optimization criteria. Each parameter has different values and dimensions; for example, the volume (V) is in cubic unit length while distance-from-road (D) is in unit length. Therefore, the cost analysis utilizes the normalized parameters ( $\hat{p}$ ), which is defined as:

$$\hat{p} = \frac{p - p_{min}}{p_{max} - p_{min}} \text{ where } 0 \leq \hat{p} \leq 1, \quad (3-17)$$

where  $\hat{p}$  is the normalized parameter,  $p$  is a parameter at the DFL, and  $p_{max}, p_{min}$  are the maximum and minimum parameters.

The minimum and the maximum value of volume (V), impact pressure (P), and distance-from-road (D) are taken from the maximum and minimum recorded values of the parameters. For the number of closed-type barriers ( $N_c$ ), the minimum number is zero (0), and the maximum number is equal to the number of initial landslide locations, i.e., source. Similarly, the minimum open-type barrier number (NO) is zero (0), and the maximum number is taken as the nearest rounded-up integer of the number of sources multiplied by 1.5.

The optimization weighting factor ( $w_i$ ) is used to consider the relative contribution of the normalized parameters. The optimization algorithm would prioritize minimizing the parameter with the highest weighting factor. For example, the optimization algorithm will prioritize minimizing the volume compared to the pressure if the weighting factor assigned to the volume and pressure are 0.5 and 0.3, respectively. To ensure the “cost” (\$) do not exceed one (1), the sum of all weighting factor should equal to one (1) and satisfy:

$$\sum_i w_i = 1, \quad (3-18)$$

where  $w_i$  is the optimization weighting factor of the normalized parameters.

The following cost function combines the normalized parameters and weighting factors to compute “cost” (\$), i.e., optimality, of a set of barrier locations:

$$\$ = \sum_i w_i \hat{p}_i \text{ where } 0 \leq \$ \leq 1, \quad (3-19)$$

where \$ is the “cost” of selected barrier locations,  $w_i$  is the optimization weighting factor, and  $\hat{p}$  is the normalized parameters.

### **3.3.5.3 Exclusion of barrier location candidates**

The debris-flow cluster locations (DFLs) close to the landslide location, i.e., the source, may collapse with the global stability failure. Furthermore, the source location is uncertain; therefore, the landslide may occur in different locations. If the expected landslides occurred in different regions, the originally proposed barrier would be ineffective.

Additionally, the candidates of suitable barrier locations are considered for the main flow channel. As multiple debris-flow clusters flow through the main flowpath channel, placing barriers along the flow channel can mitigate multiple debris-flow clusters. Therefore, the number of barriers required can be reduced by placing the barrier along the main flow channel.

Therefore, to increase efficiency, the DFLs close to the source or outside the main flow channel is excluded from the optimal barrier location selection.

## **3.4 Summary**

In this chapter, the development of an efficient and practical optimal barrier design framework has been described. The developed design framework utilizes algorithms and data structures implemented through the following design stages to select optimal barrier locations that satisfy multiple optimization criteria efficiently:

- 3D debris-flow numerical simulation with SPEC-debris,
- debris-flow cluster analysis,
- debris-flow network data structure, and
- computer-aided optimal barrier location selection with stochastic tabu search algorithm.

The method is flexible and can be utilized as a generalized method for designing optimal closed- and open-type barriers. The demonstration of the developed design framework is described in Chapter 5.

## Chapter 4. SPEC-debris-barrier Platform

### 4.1 Introduction

The common limitation of most existing debris-flow analysis methods is the lack of a platform; therefore, it is challenging to utilize the analysis methods. Furthermore, even the debris-flow analysis methods with a software package (Hung and McDougall, 2006; Pastor et al., 2009) have the following limitations: closed-source, which means the code is unviewable to the public for direct inspections; commercial, which means a purchase is required; and unobtainable, such as DAN3D (Hung and McDougall, 2006) due to licensing issues. Furthermore, the existing debris-flow simulation software does not perform additional analyses, such as risk assessment or barrier design. To overcome the current limitations of existing debris-flow simulation methods and their analysis softwares, McDougall (2016) suggested developing open-sourced software capable of performing various landslide-related analyses.

The SPEC-debris-barrier platform is an open-sourced, freely-available Windows-based software that performs the SPEC-debris model (Chapter 2) and the optimal barrier design framework (Chapters 3). The platform can simulate and design optimal barriers as mitigations against rapidly-flowing runout hazards, such as debris-flow, avalanche, and flash floods. The platform has a simple and user-friendly graphical user interface (GUI), allowing engineers inexperienced in Python3 programming to execute the developed models described in Chapters 2 and 3. Although the platform is open-sourced, the Python3 code is publicly available online; therefore, a user knowledgeable in Python3 programming can examine, modify, and utilize the developed model and framework for the optimal design of barriers against debris-flow hazards.

The SPEC-debris-barrier platform provides additional features that increase the platform's utility; for example, the platform provides the tools and options for the user to model various runout hazards and a wide range of barrier or building shapes. In addition, the platform automatically uses CPU multiprocessing to increase computational speed during analyses. Furthermore, the platform generates web-based interactive plots and animations that allow easy sharing between engineers and allows the results to be hosted on a website.

The aim of Chapter 4 is to describe the key features of the developed SPEC-debris-barrier platform, which can be used to perform combined debris-flow hazard analysis and optimal barrier design.

### 4.2 Platform Features

#### 4.2.1 Analysis overview

The SPEC-debris-barrier platform can perform various analyses by combining the following four keywords: *SPEC-debris*, *wall*, *closed*, and *combined*. The *SPEC-debris* and *wall* keywords are related to the SPEC-debris model analysis and the barrier performance analysis, respectively, as described in Chapter 2. The *closed* and *combined* keywords are the optimal barrier location selection framework described in Chapter 3. However, the *closed* keyword considers closed-type barriers only, while the *combined* keyword considers both closed-type and open-type barriers.

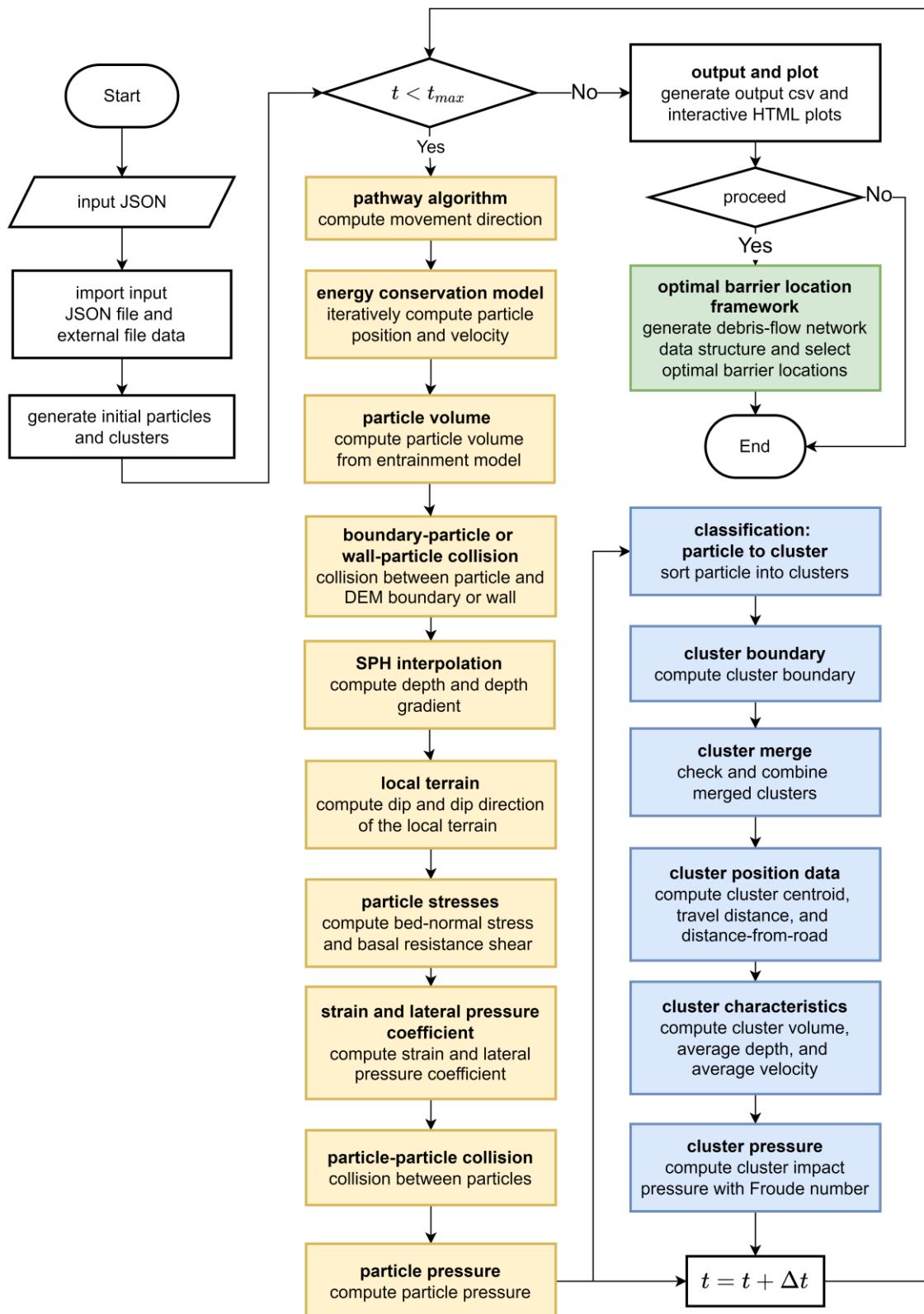


Figure 4.1 Overall flowchart of the SPEC-debris-barrier platform integrating SPEC-debris method (yellow box), debris-flow cluster analysis (blue box), and optimal barrier location framework (green box)

The six analysis keywords accepted by the SPEC-debris-barrier platform and their corresponding set of analyses are tabulated in Table 4.1.

Table 4.1 Analysis keywords and their corresponding analyses

Analysis Keyword	Runout Analysis	Optimal Barrier Location Selection		Barrier Performance Analysis
		Closed-type Only	Both Open- and Closed-type	
<i>SPEC-debris</i>	O	X	X	X
<i>SPEC-debris-wall</i>	O	X	X	O
<i>SPEC-debris-closed</i>	O	O	X	X
<i>SPEC-debris-combined</i>	O	X	O	X
<i>closed</i>	X	O	X	X
<i>combined</i>	X	X	O	X

Note: O = analysis performed, X = analysis not performed

#### 4.2.2 User Interface

The SPEC-debris-barrier platform has an intuitive and simple graphical user interface (GUI) that allows an engineer experienced in the field of runout analysis and landslide geohazards to interact with the software easily. The software automatically disables and enables the buttons to ensure that the user cannot skip an operation stage without clearing the previous step. For example, the platform prevents users from starting the analysis without checking the input file for errors. The platform has two text displays and six buttons, as shown in Figure 4.2. The six buttons and their functions are:

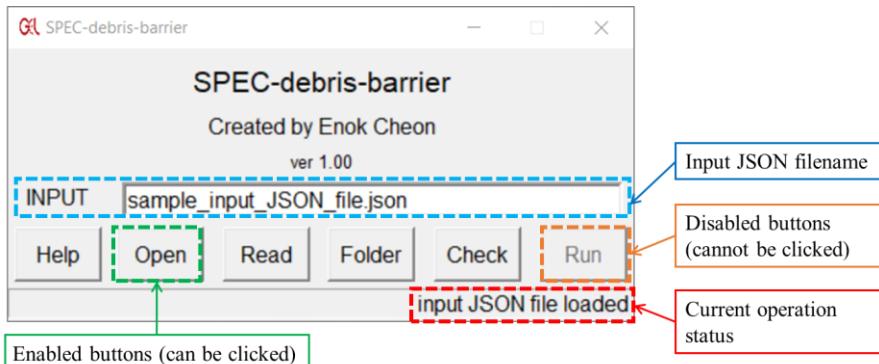


Figure 4.2 User interface of SPEC-debris-barrier platform

- (1) Help: opens the user manual,
- (2) Open: opens the file explorer and prompts the user to select the input JSON file,
- (3) Read: opens the input JSON file on a text editor,
- (4) Folder: opens the file explorer to the location of the input JSON file,
- (5) Check: performs a check for any errors in the input JSON file,
- (6) Run: opens a new command prompt (cmd) and start the analysis.

The SPEC-debris-barrier platform also features a simple checking function for the input JavaScript Object Notation (JSON) file. A pop-up window displays the error message if any error is detected during the check, as shown in Figure 4.3. The error message provides a source of error and suggests suitable corrections.

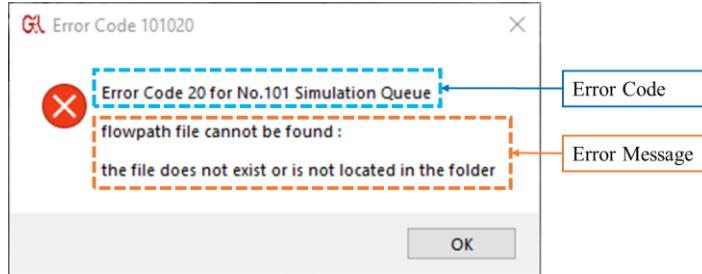


Figure 4.3 Example of an error message pop-up window in the SPEC-debris-barrier platform

#### 4.2.3 Instruction

The following procedure is used for installing and performing SPEC-debris analysis and optimal barrier location framework in the SPEC-debris-barrier platform:

- Installation
  - Download the Python3 installation program from the official website ([python.org](http://python.org)). Installing the 64-bit python version 3.9 or higher is recommended. While installing, ensure that the “Add Python 3.xx to PATH (xx is version number)” option is clicked (Figure 4.4). If Python3 is already installed and the python file location added to the PATH, this stage can be skipped
  - Although not compulsory, installing visual studio code (VS code) is highly recommended. The VS code is a code editor that can automatically detect syntax errors on the input JSON file
  - Unzip the zip file containing the platform and select the exe file to start the platform

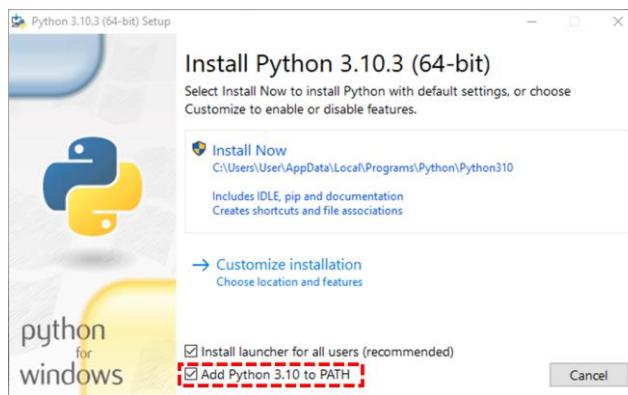


Figure 4.4 Selecting the “Add Python 3.xx to PATH” option during Python3 installation

- Starting up
  - Find the SPEC-debris-barrier platform on the start menu or desktop
  - Double click and start the software

- Select input JSON file (Figure 4.5)
  - Double click the “Open” button. All other buttons except “Help” is disabled; hence, the user cannot proceed without selecting an input JSON file (Figure 4.5a)
  - Navigate through the file explorer and select the input JSON file
  - If successful, the input JSON filename would appear in the INPUT text box, and the “input JSON file loaded” text will appear on operation status. The “Read” and “Folder” buttons would be enabled (Figure 4.5b)
  - The user can change the input JSON file in two ways:
    - If the new input JSON file is located in the same folder as the current input file, change the filename using the INPUT text box
    - If the new input JSON file is located in a different folder, double-click the “Open” button and select a new JSON file located in the other folder

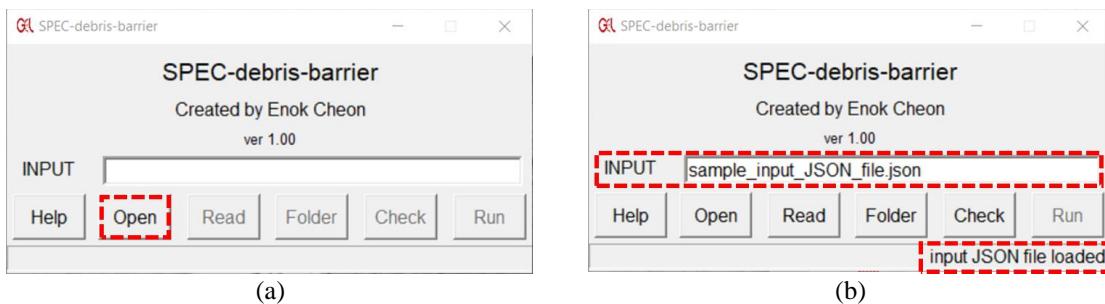


Figure 4.5 Loading input file: (a) opening input JSON file, and (b) successful loading of an input JSON file

- Viewing input JSON file (if required)
  - Double click the “Read” button opens a text editor, such as VS code, and displays the input JSON file. The user can read and edit the input file
  - Double click the “Folder” button opens the folder where the input JSON file is located, allowing easy access to input and output files
- Checking input JSON file
  - Double click the “Check” button to check for potential errors in the input JSON file
  - If an error is detected, an error message will pop up, as shown in Figure 4.3, and prompt the user to fix the sources of error. Use the “Read” button or the text editor to modify the input JSON file
  - If no error is detected, a pop-up window will inquire whether the user wants to proceed to start the analyses, as shown in Figure 4.6. To start the analysis, click the “Yes” button. If the user wishes to perform the analyses later, click the “No” button
  - If the check did not find any errors in the input JSON file, the “Run” button is enabled

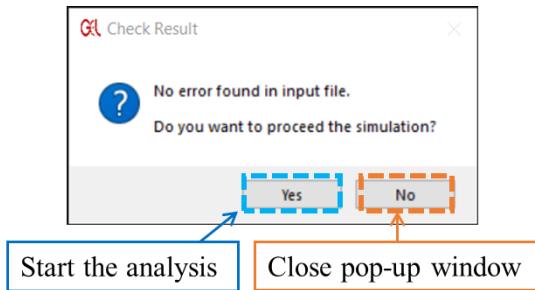


Figure 4.6 Pop-up window after checking to inquire whether to proceed to the analyses

- Perform analyses
  - Click the “Run” button to perform analyses
  - A new command prompt (cmd) will be opened. The cmd will display the completion of a particular stage in the analysis. Additionally, the SPEC-debris analysis progress will be tracked by showing the following: simulation time, computation time, and expected remaining computation time (Figure 4.7)
  - If the user wants to terminate the analyses before completion, press the “CTRL” button, then the “C” button (“CTRL + C”). The cmd displays “user terminated early” to indicate early termination of the analyses (Figure 4.7)
  - When the analyses are complete, the cmd will display a message: “Press any key to continue . . .” Press any key to close the cmd (Figure 4.7)
  - When the cmd is closed, a pop-up window opens, messaging the termination of the analyses (Figure 4.8a), and “simulation complete” text will be displayed on operation status (Figure 4.8b)

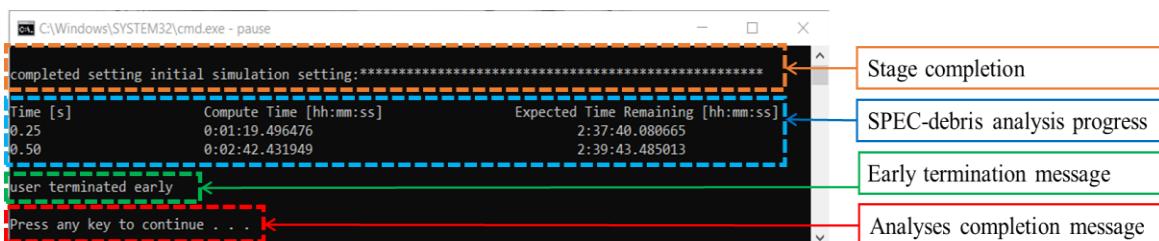


Figure 4.7 Command prompt (cmd) analyses progress message

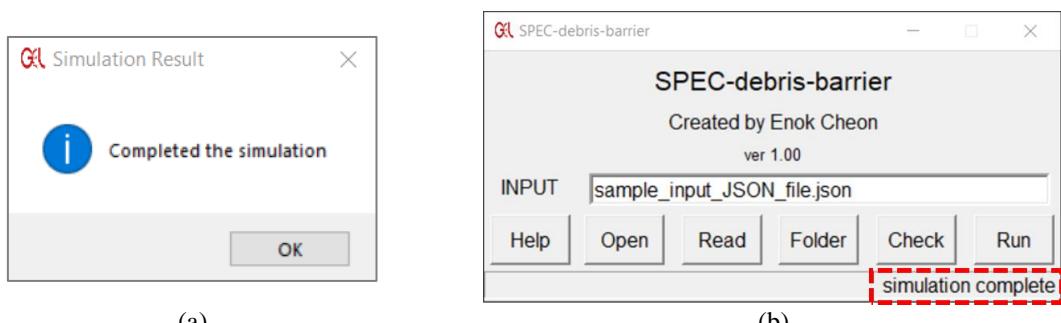


Figure 4.8 Analyses completion message on: (a) pop-up window and (b) operation status

- Accessing the output files
  - The output files are generated when the analyses are complete. Use the “Folder” button to quickly open the folder where the input and output files are located
  - Output JSON files summarize the input data and the analyses results
  - Output csv files contain computed parameters in tabulated format
  - Output HTML files are the interactive plots and animations, which can be viewed by opening the files through internet browsers, such as Google Chrome or Microsoft Edge

## 4.3 Input

### 4.3.1 JSON file format

The SPEC-debris-barrier platform uses the JavaScript Object Notation (JSON) file format to read and ascribe input data. JSON file format provides several advantages:

- (1) easy to read and understand,
- (2) variable data types can be described,
- (3) a lightweight file format that can load input data more quickly onto the platform,
- (4) Python3 can quickly load and generate JSON files,
- (5) professional code editing tools (e.g., VS code) can be used for writing JSON files, and
- (6) widely utilized in web programming; hence, it is well documented and standardized.

Therefore, a JSON format is used instead of a plain text file format to input topography data, material properties, and analysis options.

The general template for an input JSON file is shown in Figure 4.9. The user must select one of the valid analysis keywords and assign appropriate data to the input variables. The user can specify several input data sets; therefore, multiple analyses can be performed on the SPEC-debris-barrier platform using a single input JSON file. For example, suppose two input sets share the analysis keyword. In that case, the input sets are written under the same analysis keyword because the JSON file format does not allow the repetition of analysis keywords. Therefore, the input sets with the same analysis keyword are described between the square brackets ([...]) separated by a comma (,) symbol, as shown by the orange and blue input sets in Figure 4.9.

The comprehensive instructions for creating an input JSON file are listed in Appendix A.

## 4.4 Output

### 4.4.1 Analysis results: JSON

The SPEC-debris-barrier platform generates a JavaScript Object Notation (JSON) file summarizing the analysis inputs and results. Additionally, when *SPEC-debris-closed* and *SPEC-debris-combined* analyses are performed, an input JSON file can be used to perform closed and combined analyses are generated. The user can easily modify these generated input JSON files to modify the optimal barrier location selection parameters; therefore, the user would not need to re-perform the SPEC-debris analysis.

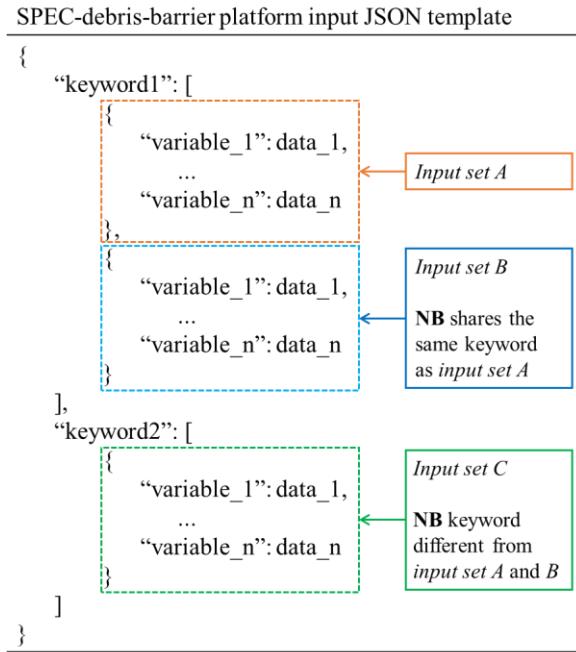


Figure 4.9 Template of the input JSON file

#### 4.4.2 Interactive HTML plots

The SPEC-debris-barrier platform generates interactive static and animation plots in two- and three-dimensions (2D and 3D) through the Python3 plotly library. Unlike the other runout analysis software producing static figures and animation videos, the platform generates interactive plots in an HTML file format, which provides several advantages over plots in image or video file formats:

- easily viewable through any electronic device with an internet browser without requiring neither third-party software nor an internet connection
- user-friendly features allowing interactions (zoom, rotate, inspect, etc.) with the plot
- high-quality plots with no loss in data or plot resolution
- easily hostable on any internet websites

##### 4.4.2.1 Scatter, heatmap, and animation in 2D and 3D

The interactive static plots summarize the overall debris-flow simulation results by displaying the propagation of the debris-flow clusters and particles over a 3D topography in a single frame. The 3D topography is plotted in the background as a contour and a surface plot in 2D and 3D static plots, respectively (Figure 4.10). In the 2D contour plot, a number is marked near the grey lines to denote the topography elevation. In the 3D surface plot, the elevation is presented with a continuous colormap between brown and teal colors representing high and low elevations.

The debris-flow clusters propagation is presented as a scatter plot with a circle (2D) or a sphere (3D) marker connected by straight lines. As shown in Figures 4.11(a) and 4.11(b), the markers represent the centroid location of the clusters at a particular time step, and the lines connecting the markers show the propagation of

cluster centroid locations over the time step. As shown in Figures 4.11(c) and 4.11(d), the cluster circle and sphere markers are colored to convey the following cluster parameters: cluster id number, average depth ( $h$ ), average velocity ( $u$ ), cumulative volume ( $V$ ), average impact pressure ( $P$ ), and average distance-from-road ( $D$ ). The connecting straight lines are colored according to the cluster id number; hence, the viewer can easily track the propagation of a particular debris-flow cluster and distinguish different clusters.

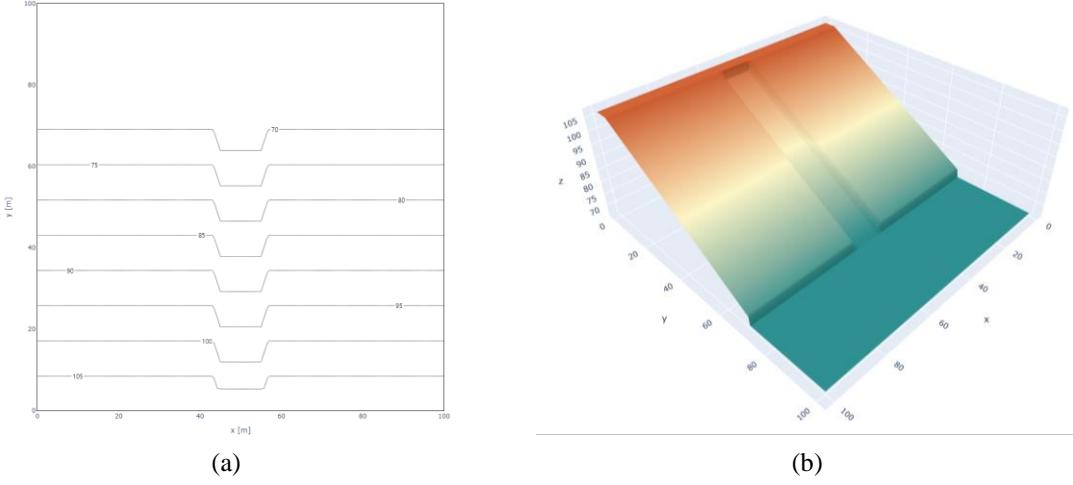


Figure 4.10 Topography plot in (a) 2D as contour maps and (b) 3D as a surface plot

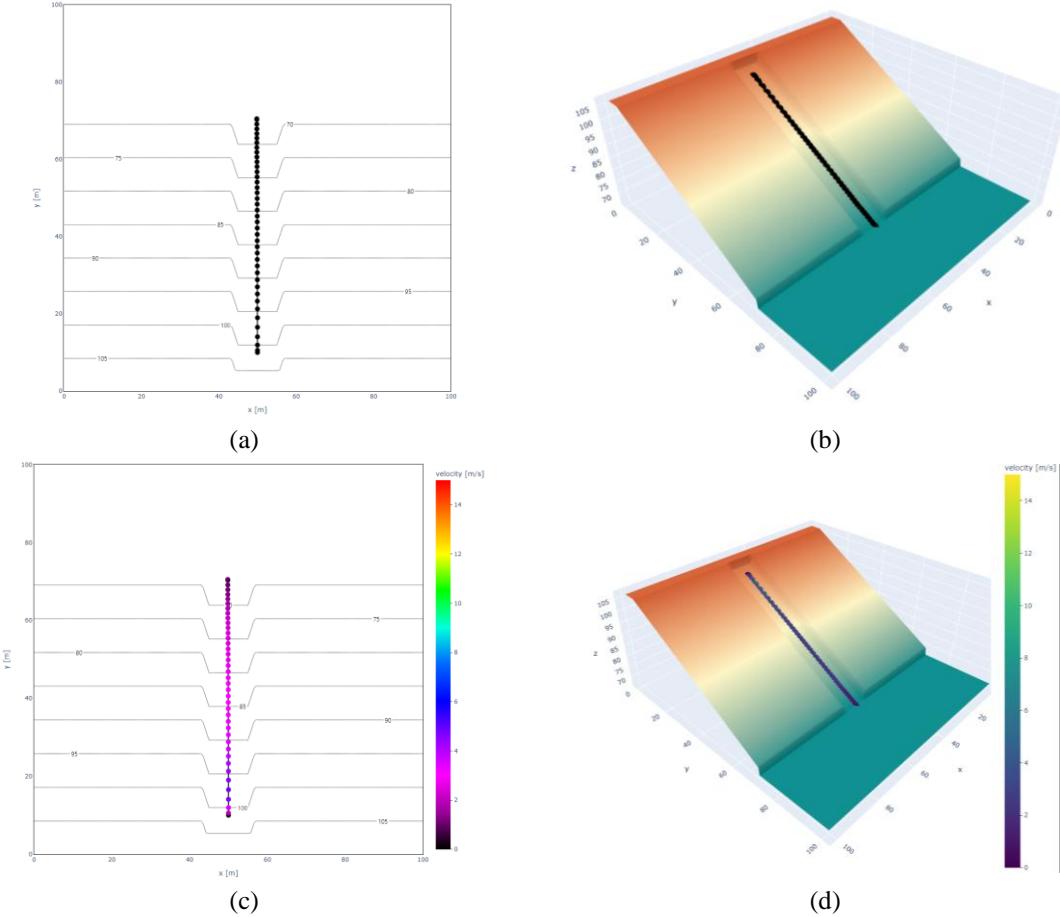


Figure 4.11 Debris-flow cluster static plot examples: (a) cluster flowpath in 2D, (b) cluster flowpath in 3D, (c) average cluster velocity in 2D, and (d) average cluster velocity in 3D

Due to the high number of debris-flow particles in the SPEC-debris simulation, a single static plot is insufficient to present every location and parameter of the particles across the entire simulation. Instead, the platform generates heatmap plots that color each cell according to maximum particle parameters seen by each cell since the beginning of the simulation. The following particle parameters are tracked for the heatmaps: depth ( $h$ ), velocity ( $u$ ), volume ( $V$ ), and impact pressure ( $P$ ). If cluster analysis (refer to Section 3.3.3) is not performed, only the particle heatmap is generated, as shown in Figure 4.12(a). Otherwise, the cluster scatter plot is overlayed on top of the particle heatmap plots, as shown in Figure 4.12(b).

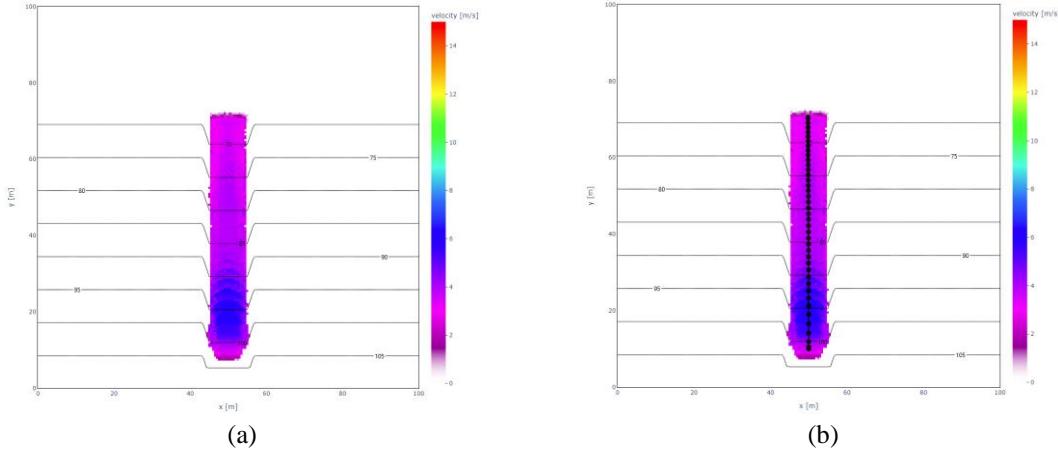


Figure 4.12 Examples of maximum recorded debris-flow particle velocity heatmap plot with  
(a) only the particle results and (b) both particle and cluster results

A single static plot is ineffective for presenting the debris-flow particle simulation results; therefore, the platform generates the scatter animations. The particle positions are updated with each time step. As shown in Figure 4.13(a), the particles are displayed as a circle (2D) or a sphere (3D) marker. They are colored to convey the following particle parameters: cluster id number, depth ( $h$ ), velocity ( $u$ ), volume ( $V$ ), and impact pressure ( $P$ ). Similar to the particle heatmap plots, the cluster scatter plot is overlayed on top of the particle animations, as shown in Figure 4.13(b).

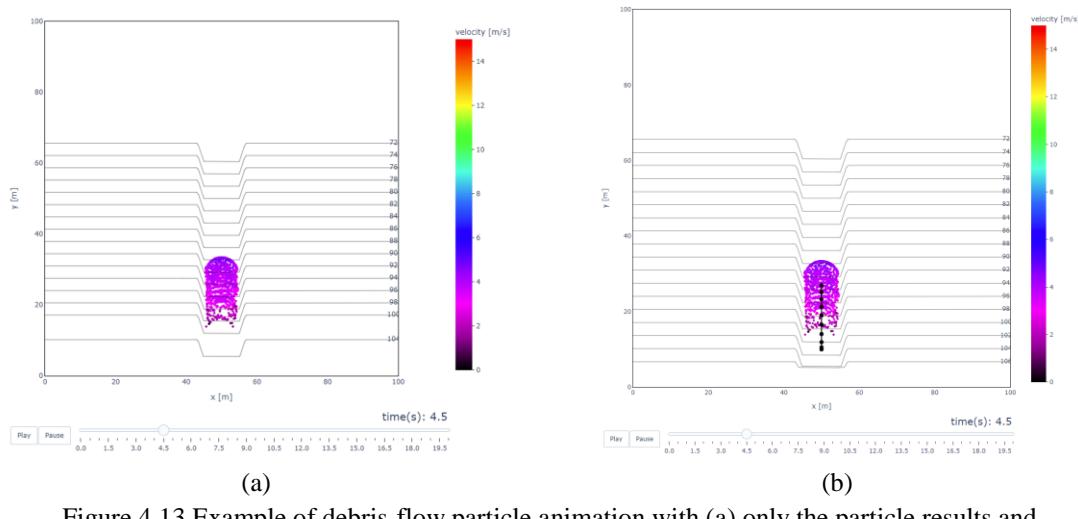


Figure 4.13 Example of debris-flow particle animation with (a) only the particle results and  
(c) both particle and cluster results

#### 4.4.2.2 Interactive features

All HTML static plots and animation generated by the SPEC-debris-barrier platform provide several features for users to interact with the plot for closer examination (Figure 4.14):

- trace – provides plot details where the mouse cursor points,
- export – save the current plot/frame as an image,
- zoom – select region to zoom,
- pan – move the plot around with the mouse,
- rotate – rotate plot around an axis (only in 3D),
- default view – reset the plot orientation to be in the default view, and
- playback – play and navigate through animation frames.

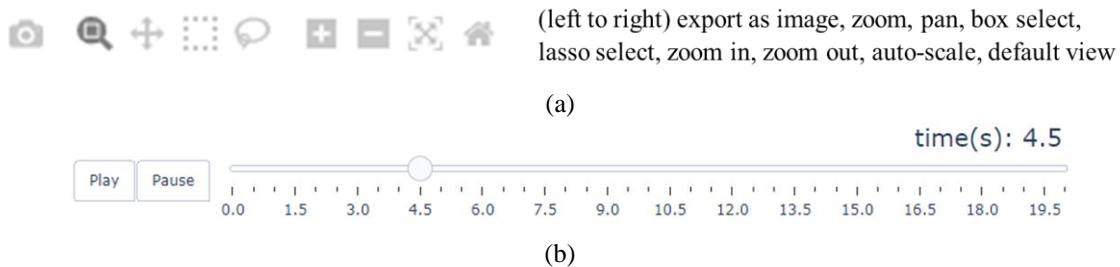


Figure 4.14 Interactive plot features: (a) in-built feature buttons and (b) playback features for animations

The user can perform the following interactive features with the mouse and keyboard instead of selecting one of the in-built features buttons placed at the top-right corner:

- rotate image – (1) right-click or (2) SHIFT + right-click,
- pan image – (1) left click or (2) CTRL + right-click,
- zoom in – scroll wheel upwards, and
- zoom out – scroll wheel downwards.

## 4.5 Discussion and Conclusion

The SPEC-debris-barrier platform described in this chapter is developed to solve runout analysis across 3D terrains, search for optimal barrier locations, and compute the performance of various closed and open barriers.

The following features of the developed platform distinguish itself from other existing runout hazard and barrier performance analyses software:

- comprehensive computation tool that can conduct for both analyzing rapidly-flowing runout hazards and designing barriers for mitigation against runout hazards,
- versatile modeling capabilities for buildings and barriers, such as closed-type, slit-type, reflective type, and circular structures (refer to Appendix A),
- interactive web-based HTML static and animation plots with in-built functions,

- intuitive and simple graphical user interface (GUI) for users to quickly learn and use the platform,
- utilizes multiprocessing of CPUs for faster computation (refer to Appendix A),
- open-sourced freely available software,
- input and output data transfer through JSON and csv files; hence the platform can be used as the backend of the runout analysis website.

As a computational tool, the results from the SPEC-debris-barrier depend on the specific topography data, initial starting location, material properties, and optimization criteria the user provides. Therefore, an experienced engineer in runout analysis and landslide geohazards should carefully examine and verify the input and output data based on their engineering judgment or other analyses.

## Chapter 5. Tests and Case Studies

### 5.1 Introduction

The SPEC-debris-barrier platform, which contains the SPEC-debris runout simulation method and the optimal barrier design framework, has been introduced in Chapters 2, 3, and 4. The SPEC-debris simulation method incorporates essential principles and methodologies for debris-flow hazard assessment and barrier performance analysis. However, it is essential to verify the accuracy of the developed simulation method. Therefore, this chapter aims to validate and evaluate the SPEC-debris-barrier platform by comparison with experiments and case studies.

The accuracy and applicability of the SPEC-debris method for debris-flow simulation and barrier performance were verified through the case studies of the 2011 debris-flow incident at Mt. Umeyeon and small-scale flume laboratory experiments, respectively. The applicability of the developed optimal barrier design framework for designing optimal closed and open-type barriers was tested through a parametric study conducted on Mt. Umeyeon.

### 5.2 Barrier performance

#### 5.2.1 Description

Choi et al. (2018) performed a small-scale flume experiment to assess the effects of slit-type barriers on debris-flow characteristics. A slit-type barrier is an open-type barrier with spacing between a series of walls. The arrangements of slit-type barriers tested in the experiment are shown in Figure 5.1. The experimental setup is shown in Figure 5.2.



Figure 5.1 P00 slit-type barrier

The debris-flow material comprised a sand-water mixture with 1 kg of Jumunjin sand, uniformly-sized dry sand with  $D_{50}$  of 0.6mm, and 3.5 kg of water. The sand-water mixture was placed at the uppermost section of the flume behind a gate. At the start of the experiment, the gate was opened to let the sand-water mixture flow.

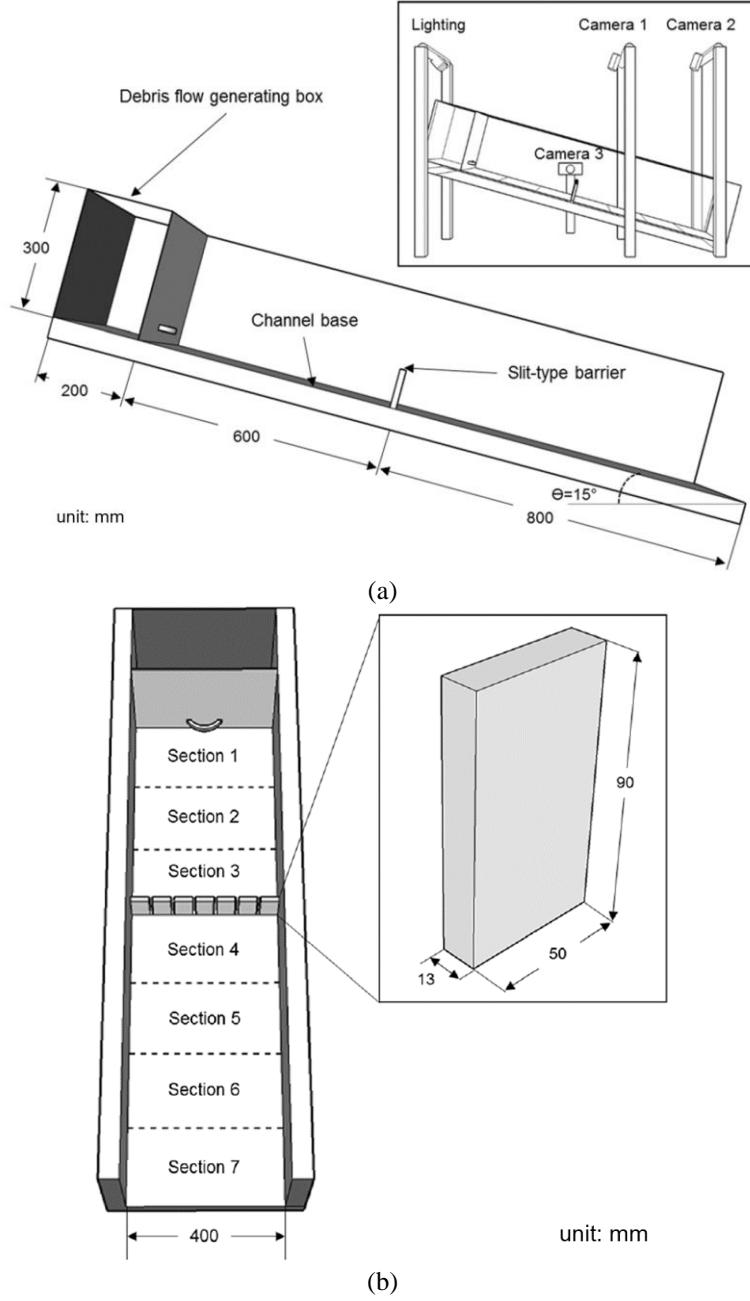


Figure 5.2 Small-scale flume experimental setup: (a) section view and (b) plan view (Choi et al., 2018)

### 5.2.2 Methodology

The flume laboratory experiment was recreated in the SPEC-debris-barrier platform in a 1:1 scale model. The digital elevation model (DEM) grid spacing was 10 mm. The number of particles per cell was set as one (1); therefore, the total number ( $N$ ) and the radius ( $R$ ) of particles ( $N$ ) were 760 and 5 mm, respectively. As the soil-water mixture was confined within the channel walls and the gate, the *initial SPH* variable was set to false so that the SPEC-debris directly uses the depth value given by the source input csv file. The time step interval of 0.02s was selected, and a maximum simulation time of 1.0s was performed.

Using the specific gravity ( $G_s$ ) = 2.65, the density of the sand-water mixture was set as 1160.6 kg/m<sup>3</sup>. The internal rheological friction angle was set to  $\phi_i$  = 32° based on a back-analysis performed with FLOW-3D in Kim et al. (2020) paper. As the flume channel has no erodible soil, no entrainment phenomena occurred; hence, the entrainment growth rate ( $E_s$ ) value was set to zero (0).

Based on the descriptions from Choi et al. (2018), the *wall info* variable set in the input JSON file for each slit-type barrier shown in Figure 5.1 was the following:

Table 5.1 P00 slit-type barrier model input

Slit-type barrier	P00
wall type	“P”
slit ratio	0.125
orientation of wall segments (Polar)	0.0
orientation of wall overall (Polar)	0.0
number of wall segments	7
wall thickness	0.02
wall length	0.4
wall elevation option	2
elevation or height	0.09
wall centroid X	0.2
wall centroid Y	0.77

Units: angle in degree°; length and coordinate position in meter

The CFD-DEM simulation by Li and Zhao (2018) utilized the coefficient of restitution (COR) to simulate the energy loss due to collision. In Li and Zhao's (2018) paper, the COR between debris-flow and flexible barrier (COR\_p2w) and between particles (COR\_p2p) values were set as 0.3 and 0.7, respectively. Using Li and Zhao's (2018) COR coefficients as the initial values, the COR coefficients were modified through trial-and-error to match the observed P00 slit-type barrier performance.

### 5.2.3 Results and discussion: barrier performance

The simulation result with no barrier installed, i.e., reference case, is shown in Figure 5.3. The model provided a good match of the observed data with the following basal rheological parameters: basal frictional coefficient ( $f_b$ ) = 0.2 and turbulence coefficient ( $f_t$ ) = 0.01, which are equivalent to the basal friction angle of  $\phi_b$  = 11.3° and turbulence resistance coefficient  $\xi$  = 100 m/s<sup>2</sup>. The simulated frontal velocity of 2.91 m/s matches the observed frontal velocity of around 2.90 m/s at the Section 3 location (Figure 5.2(b)).

The same basal rheological parameters from the reference case were used to evaluate the performance of the P00 slit-type wall through the SPEC-debris method. The effect of the barrier is shown in Figure 5.4. In Choi et al. (2018), the barrier performance was measured with the trap ratio [=  $(V - V_{ref})/V_{total}$ ] and velocity reduction [=  $(u_{barrier} - u_{ref})/u_{ref}$ ] at Section 3 location (Figure 5.2(b)). The observed trap ratio and velocity reduction for P00 slit-type barriers were 19% and 42%, respectively.

The simulated results had a good agreement when COR between debris-flow and flexible barrier (COR\_p2w) was 0.7 and COR between particles collisions (COR\_p2p) was 0.6. The computed trap ratio and velocity reduction by SPEC-debris for P00 slit-type barriers were 23% and 44%, respectively. Both performance metrics are within a 5% difference from the observed results.

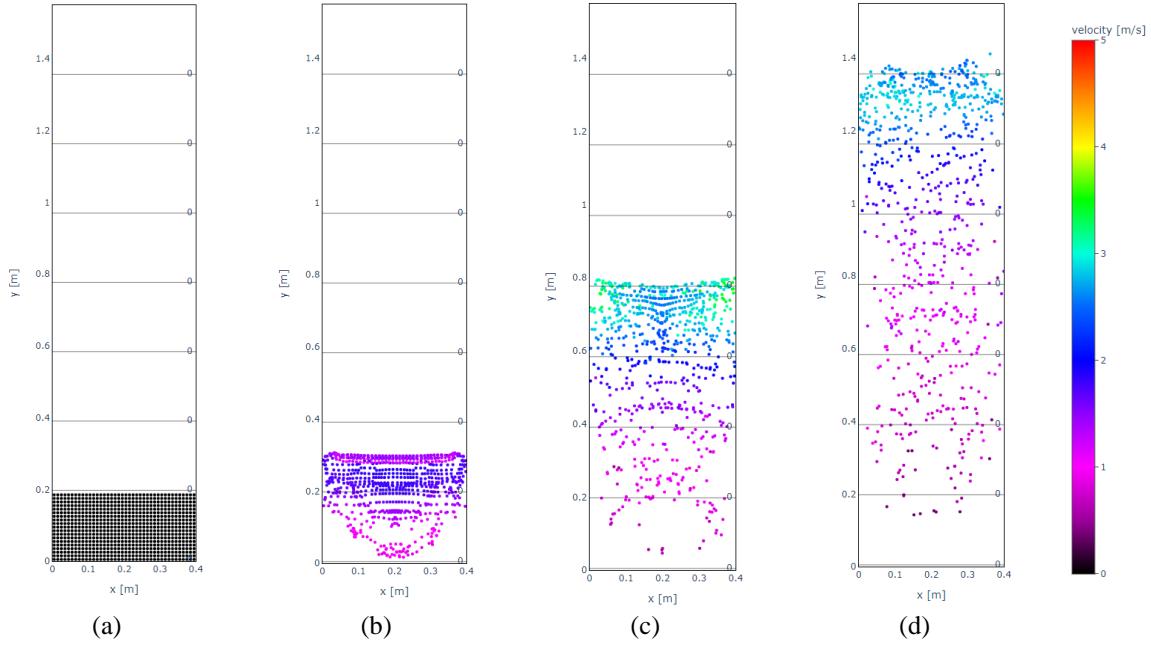


Figure 5.3 SPEC-debris simulation snapshots of reference case:

(a)  $t = 0$ , (b)  $t = 0.2\text{s}$ , (c)  $t = 0.4\text{s}$ , and (d)  $t = 0.6\text{s}$

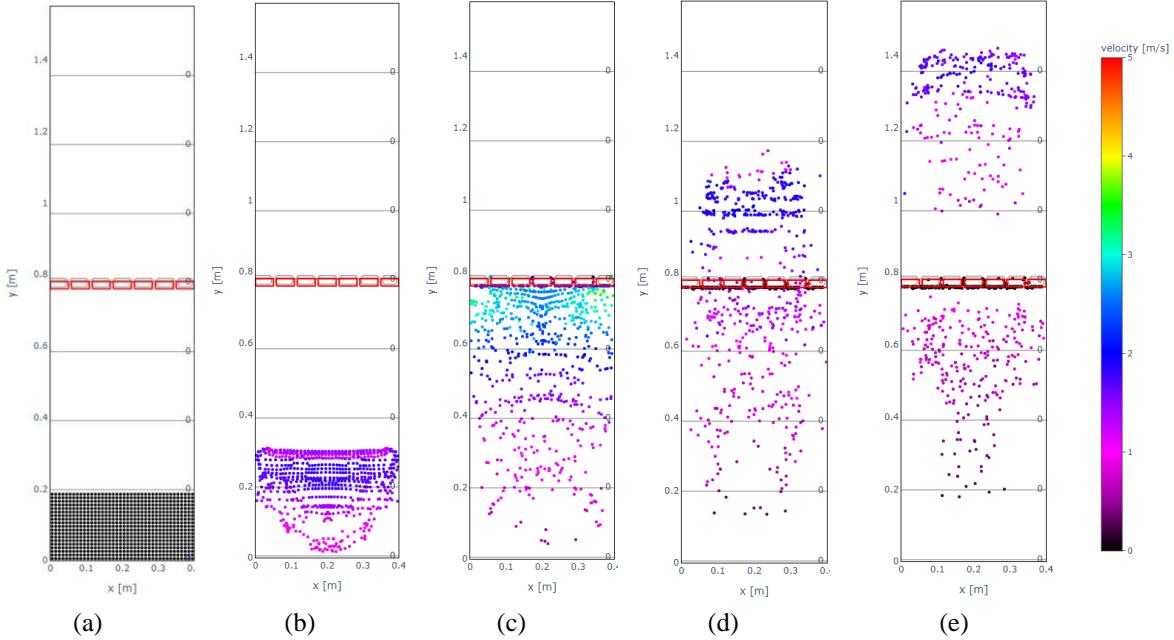


Figure 5.4 SPEC-debris simulation snapshots of P00 slit-type wall:

(a)  $t = 0$ , (b)  $t = 0.2\text{s}$ , (c)  $t = 0.4\text{s}$ , (d)  $t = 0.6\text{s}$ , and (e)  $t = 0.8\text{s}$

Additionally, Kim et al. (2020) evaluated the barrier performance for the same flume experiment using FLOW-3D (Hirt and Nichols, 1988). The Kim et al. (2020) paper evaluated the barrier performance with volume reduction [=  $V_{\text{barrier}}/V_{\text{total}}$ ] and velocity reduction [=  $(u_{\text{barrier}} - u_{\text{ref}})/u_{\text{ref}}$ ] at Section 3 location (Figure 5.2(b)). The volume ratio and velocity reduction for the P00 slit-type wall computed by FLOW-3D were 36.3% and 44.0%, respectively. The developed SPEC-debris method closely matched the results of the FLOW-3D by computing 31.4% and 44.0% for the volume ratio and velocity reduction. Therefore, the SPEC-debris method has been shown to accurately evaluate open-type barrier performance matching the results of a 3D CFD simulation method.

#### 5.2.4 Results and discussion: particle number per cell

As described in Appendix A, the SPEC-debris numerical model generates the initial debris-flow particles from the source data file, which specifies the initial landslide failure mass location and depth. The *particle number per cell* variable specifies the number of particles generated per cell. As shown in Figure A.4, the cell is subdivided based on the *particle number per cell* value, and a circular particle is fitted inside the subdivided cell. Through this process, the SPEC-debris model can increase the number of particles and decrease the size of the particles.

The effect of *particle number per cell* value on the barrier performance has been studied through a parametric study. The same flume model (Figure 5.2), P00 slit-type barrier (Figure 5.3), and material properties were used. The *particle number per cell* values of 1, 4, and 9 were analyzed. The computational time was tested with AMD Ryzen 9 3950X CPU with CPU multiprocessing of 20 threads. The following barrier performance metric was used to check the effect of the *particle number per cell* value, and the results are summarized in Table 5.2:

$$\text{Velocity Reduction} = \frac{u_{\text{with barrier}} - u_{\text{without barrier}}}{u_{\text{without barrier}}} \quad \text{and} \quad \text{Volume Reduction} = \frac{V_{\text{with barrier}}}{V_{\text{initial}}}$$

Table 5.2 Barrier performance results based on *particle number per cell*

<i>particle number per cell</i>	Number of particles (N)	Particle radius (R) [mm]	Velocity Reduction (%)	Volume Reduction (%)	Computation Time [min]	Increase in Computation Time
1	760	5	43.7	31.4	134	x1
4	3040	2.5	43.6	34.3	487	x3.6
9	6840	1.667	44.8	33.2	1145	x8.5

Note: The increase in computation time is based on the computation time of *particle number per cell* = 1

The results show that the effect of *particle number per cell* value was relatively small on the barrier performance. In order to test whether the barrier performance analysis is independent of the *particle number per cell* value, analyzing more case studies and experiments would be required in future research. At present, it is therefore recommended to use *particle number per cell* value of either one (1) or four (4) for evaluating barrier performance while being computationally efficient.

### 5.2.5 Results and discussion: barrier effect

DAN3D (Hung and McDougall, 2009) is a popular depth-averaged debris-flow simulation method that does not account for the energy loss due to collision. Unpublished research by Dr. Deuk-Hwan Lee showed that DAN3D could not verify the effect of a barrier. The barrier effect on the debris-flow propagation was evaluated with DAN3D on the same P00 slit-type barrier flume experiment (Choi et al., 2018). The DAN3D results showed that the front speed of debris-flow reduced immediately after colliding with the barrier but increased rapidly afterward. However, the developed SPEC-debris method successfully shows a decrease in the speed ratio [ $= u_{\text{with barrier}}/u_{\text{without barrier}}$ ] after collision with the P00 slit-type barrier (Figure 5.5).

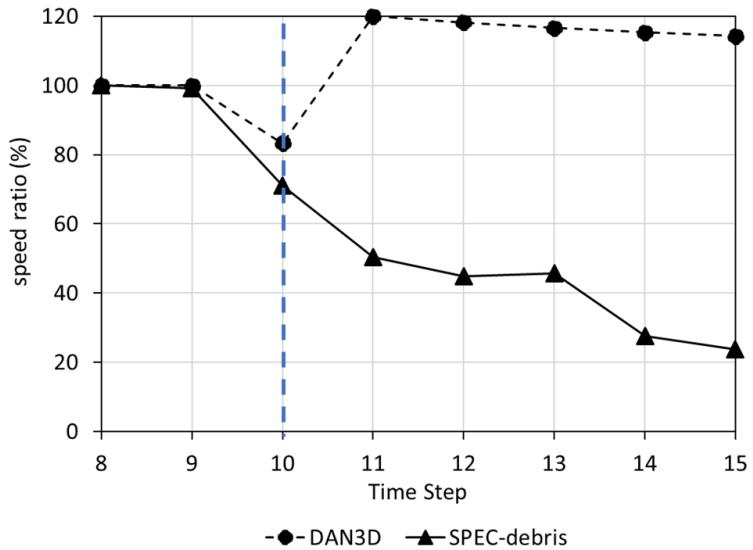


Figure 5.5 Reduction of debris-flow front speed after the moment of collision (marked by blue dash line)

The energy loss of particles after a collision with a barrier can be seen clearly in the 3D animation plot. As shown in Figure 5.6, the velocity of the particle highlighted with the red box decreased after collision with the P00 slit-type wall barrier. Therefore, unlike the DAN3D simulation method, the SPEC-debris method can successfully simulate the loss in kinetic energy, i.e., decrease in debris-flow velocity, from collisions.

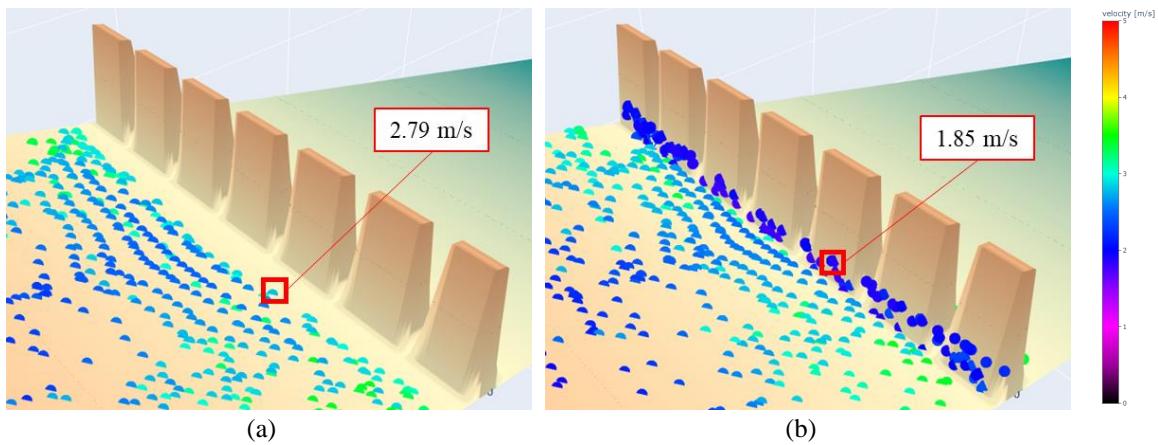


Figure 5.6 Particle collision against P00 slit-type wall: (a)  $t = 0.38\text{s}$  and (b)  $t = 0.40\text{s}$

## 5.3 Case Study: Mt. Umyeon

### 5.3.1 Description

On 26-27 July 2011, extreme rainfall with maximum intensity reaching 112.5 mm/hour and cumulatively rainfall of 470 mm triggered 147 shallow landslides and debris-flow at Mt. Umyeon in Seoul, South Korea. The geohazard caused 16 human fatalities, 20 non-fatal casualties, and extensive damage to 116 buildings in nearby regions (Yune et al., 2013). Overall, the disaster cost about 22 billion KRW for recovery and restoration.

One of the regions affected by the 2011 Mt. Umyeon disaster was the Raemian and Sindonga apartment complex, as shown in Figure 5.7. Based on the video footage, the observed velocity of debris-flow at the base of Mt. Umyeon was recorded as 28 m/s and 18 m/s for Raemian and Sindonga regions. Additionally, the depth of debris-flow at the mountain base for the Raemian and Sindonga regions is estimated to be 3 ~ 4m and 2 ~ 3m, respectively (Lee et al., 2019).

In the Raemian region, the observed velocity was around 28 m/s, around 100 kph; hence, there would have been insufficient time for evacuation in the Raemian region. However, the usage of barriers would have been suitable as a mitigation method against the debris flow in this region. Hence, this study selected the debris flow that affected the Raemian region to demonstrate the applicability of the developed optimal barrier design framework.

### 5.3.2 Methodology

The DEM of the Raemian and Sindonga region was modeled in the SPEC-debris-barrier platform with a spatial resolution of 1m. For the Raemian region, the number of particles per cell was set as one (1); therefore, the total number (N) and the radius (R) of particles (N) were 903 and 0.5 m, respectively. However, for the Sindonga region, the number of particles per cell was set as four (4), which makes the total number (N) and the radius (R) of particles (N) equal to 1428 and 0.25 m, respectively. The number of particles per cell was increased from one to four to ensure the number of particles exceeded 500.

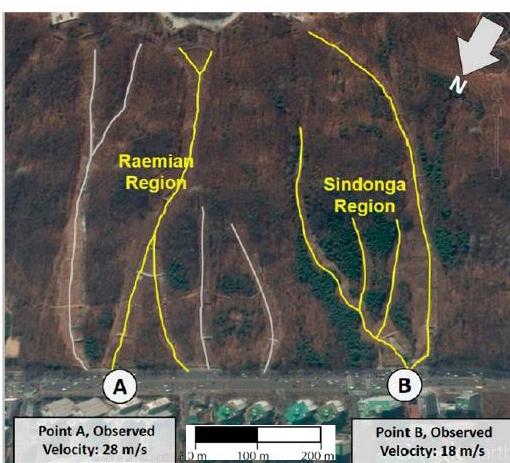


Figure 5.7 Debris-flow that affected the Raemian and Sindonga apartment complex (Lee et al., 2019)

The basal rheological parameters, the basal frictional coefficient ( $f_b$ ) and the turbulence coefficient ( $f_t$ ), for the Raemian and Sindonga regions were derived by performing back-analyses. Hungr's exponential entrainment model was selected for the analyses. Other properties were assigned to the *material* variable based on the debris-flow from Lee et al. (2019):

Table 5.3 Material properties for Mt. Umeyeon simulation

Mt. Umeyeon Region	Raemian	Sindonga
density [kg/m <sup>3</sup> ]	1885.83	1845.69
internal friction angle ( $\phi_i$ ) [°]	22.4	25.3
entrainment growth rate ( $E_s$ ) [/m]	0.00421	0.00808
maximum erodible depth [m]	4.0	4.0

A parametric study was performed on the Mt. Umeyeon Raemian region to analyze the effects of optimization criteria weighting factors on the optimal barrier locations. Two types of barrier design scenarios were considered. First, only the closed-type barrier was installed. Secondly, both closed-type barriers and open-type barriers were considered. The following optimization criteria weighting factors were selected:

Table 5.4 Optimization criteria weighting factors

Barrier consideration	Case	Volume (V)	Pressure (P)	Distance-from-road (D)	Number of closed-type barriers (Nc)	Number of open-type barriers (No)
closed-type barriers only	Case 1	0.25	0.25	0.25	0.25	
	Case 2	0.4	0.3	0.1	0.2	
	Case 3	0.1	0.1	0.4	0.4	
both open and closed-type barriers	Case 4	0.2	0.2	0.2	0.2	0.2
	Case 5	0.4	0.3	0.1	0.15	0.05
	Case 6	0.1	0.1	0.4	0.3	0.1

### 5.3.3 Results and discussion: debris-flow analysis

The back-analysis has verified the applicability of the developed SPEC-debris method for simulating fast-moving debris-flow hazards. The results of the SPEC-debris simulation on Mt. Umeyeon at the Raemian and Sindonga regions are shown in Figures 5.8 and 5.9. The comparison between the observed and computed values is listed in Table 5.5. The computed values listed in Table 5.5 are the largest value of debris-flow particles shown at the reference point A and B (see Figure 5.7). The computed value generally produces a more conservative result than the observed result.

For the Raemian region, the basal frictional coefficient ( $f_b$ ) = 0.05 and turbulence coefficient ( $f_t$ ) = 0.0 produced the best matching results to the observed data. Although the SPEC-debris method uses Voellmy's (1955) rheology, the currently well-matching model is a Coulomb frictional rheological model due to turbulence resistance being a zero value. On the other hand, for the Sindonga region, the basal frictional coefficient ( $f_b$ ) = 0.03 and turbulence coefficient ( $f_t$ ) = 0.0001 matched well with the observed results.

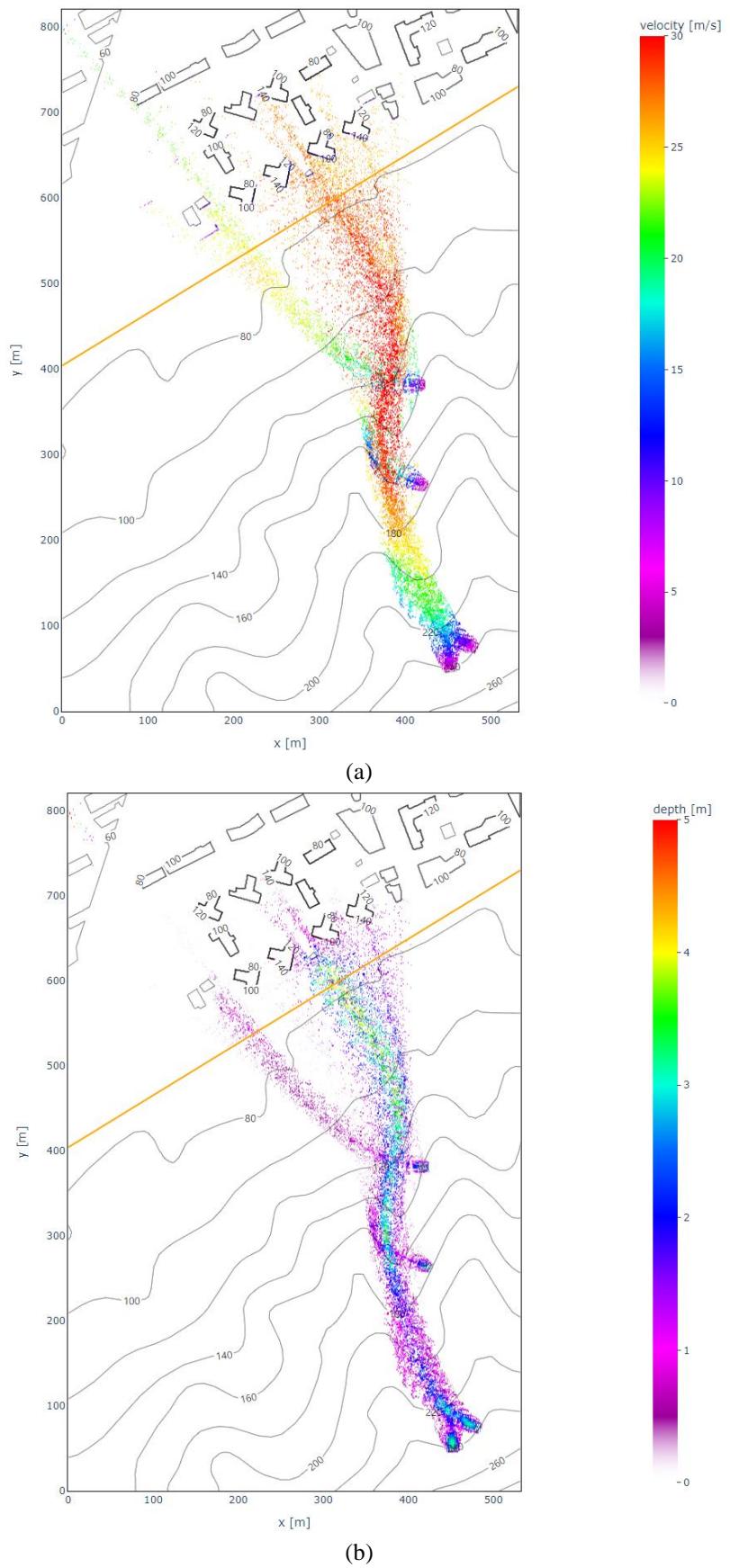
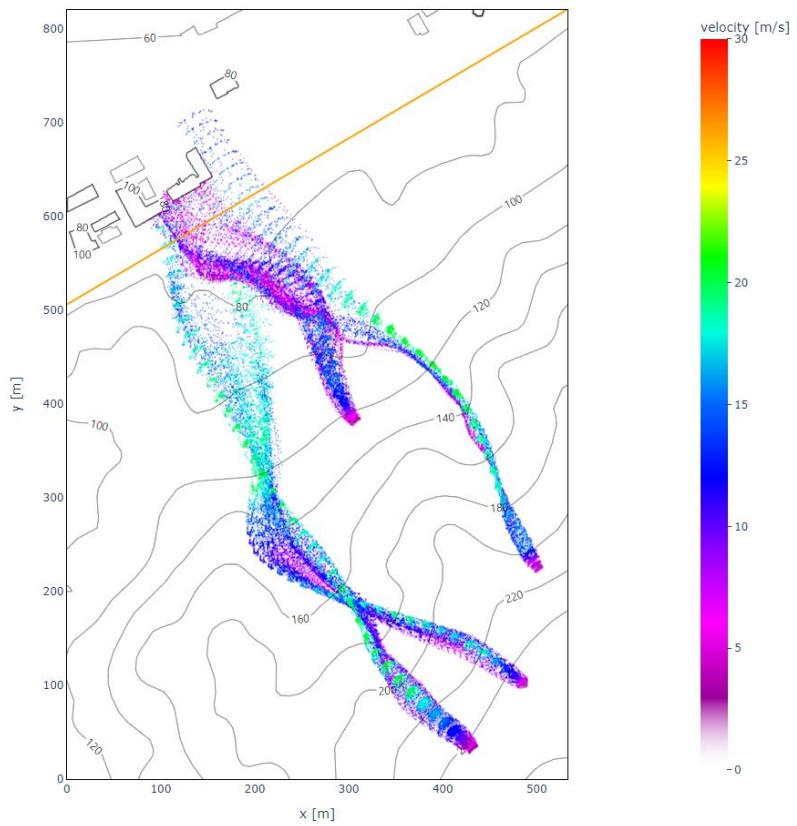
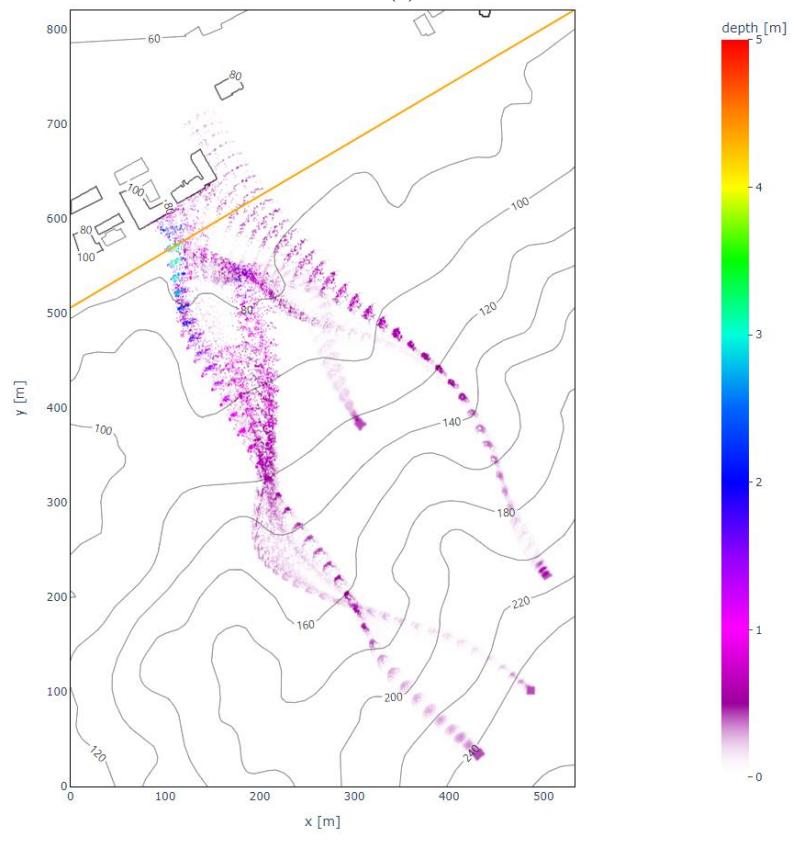


Figure 5.8 Debris-flow simulation results for the Raemian region: (a) max velocity and (b) max depth



(a)



(b)

Figure 5.9 Debris-flow simulation results for the Sindonga region: (a) max velocity and (b) max depth

Table 5.5 Comparison of simulation and observed value for Mt. Umyeon

Mt. Umyeon Region	Raemian – Point A		Sindonga – Point B	
	Observed	Computed	Observed	Computed
velocity [m/s]	28	28.8	18	17.7
depth [m]	3 ~ 4	4.1	2 ~ 3	3.4

### 5.3.4 Results and discussion: optimal closed-type barrier

A sensitivity analysis was performed on the SPEC-debris simulation results on the Mt. Umyeon to determine the effect of optimization weighting factors on selecting optimal closed-type barrier locations. The optimization criteria weighting factors from Case 1 to 3 (Table 5.4) were used.

The optimization weighting factors show the relative emphasis given to each factor. In Case 1, all factors were equal; hence, all were equally considered. For Case 2, the volume (V) and pressure (P) were prioritized so that the optimal barrier location would minimize the barrier size and reinforcement. For Case 3, priorities were given to the distance-from-road (D) and the number of barriers (Nc); hence, the optimal barrier location would maximize the accessibility and minimize the environmental impact of the barriers.

The sensitivity analysis shows that all cases (Cases 1 to 3) selected the exact locations as their optimal barrier locations, as shown in Figure 5.9. The volume, impact pressure, and the distance-from-road on each selected location in Figure 5.9 are summarized in Table 5.6.

Table 5.6 Debris-flow cluster characteristics at each optimal closed-type barrier location

Optimal closed-type barrier locations	Volume		Pressure		Distance-from-road	
	Computed [m <sup>3</sup> ]	Normalized	Computed [kPa]	Normalized	Computed [m]	Normalized
A	1816.8	0.0704	234.7	0.3411	214.6	0.4026
B	2403.8	0.0931	320.6	0.4660	484.7	0.9094
C	862.6	0.2125	226.2	0.3921	101.6	0.1631
D	424.5	0.1046	249.7	0.4328	409.7	0.6575
E	314.8	0.0776	229.1	0.3972	449.4	0.7212
F	1128.9	0.278	304.0	0.5270	276.2	0.4432

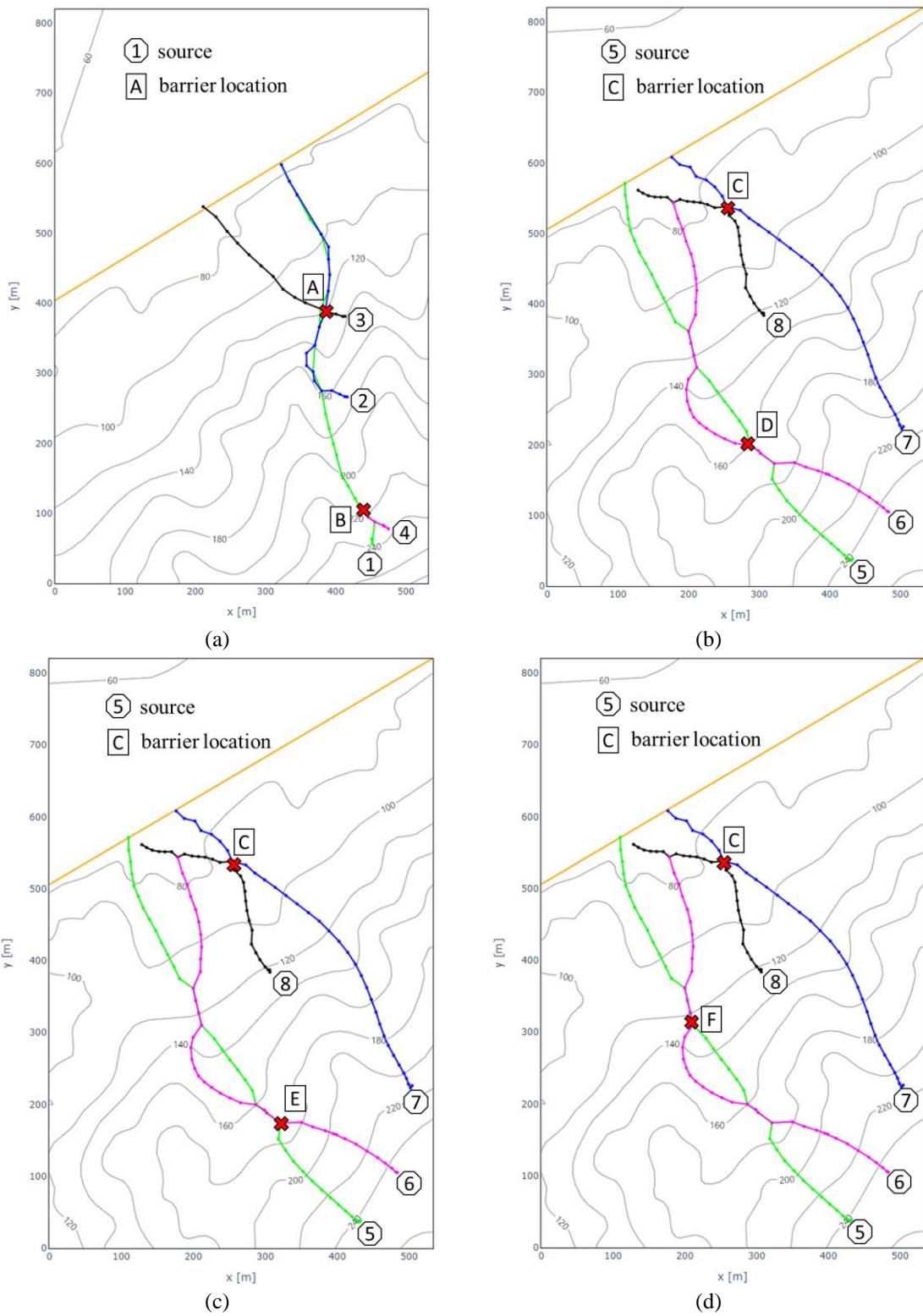


Figure 5.10 Optimal closed-type barrier locations (marked by the red cross and

the road marked by the yellow line) based on the optimization criteria weighting factor:

(a) Raemian – Cases 1, 2, and 3; (b) Sindonga – Cases 1; (c) Sindonga – Cases 2; and (d) Sindonga – Cases 3

The explanation for invariance in optimal closed-type barrier locations at Mt. Umyeon Raemian region with the optimization criteria weighting factors would be the resultant of cluster volume and pressure at the base. For example, suppose the distance-from-road (D) was minimized in Case 3 by placing the closed-type barriers near the road (marked by a yellow line in Figure 5.10(a)). In that case, the exponentially-increased volume from the entrainment phenomena and the rapid speed (28 m/s) reaching the toe of the slope would have significantly increased the cost (\$). Consequently, the optimal closed-type barrier location selection prioritized minimizing the impact of debris-flow volume and pressure by placing the barriers close to their source locations. The barrier location A mainly blocks the debris-flow from sources 2 and 3, while the barrier location B prevents the propagation of debris-flow from sources 1 and 4.

For the Mt. Umyeon Sindonga region, the optimization criteria weighting factors only affected one closed-type barrier location. The barrier location C, which blocks the debris-flow from sources 7 and 8, did not change with the weighting factors cases. However, the location of the closed-type barrier that blocks the debris-flow from sources 5 and 6 changed depending on the weighting factors cases.

In general, the debris-flow volume is lower near the source as the effect of entrainment phenomena is proportional to the distance traveled from the source. For Case 2, which prioritizes minimizing the volume and impact pressure of debris-flow, the barrier location E was the ideal location to block debris-flow from sources 5 and 6 with the smallest volume. In contrast, Case 3 prioritizes minimizing the distance-from-road; therefore, the barrier location F, which is closer to the road (marked by a yellow line in Figure 5.10), was selected. Finally, as Case 1 gives equal consideration to all the design elements, barrier location D, which is located between the barrier locations E and F, was selected.

Overall, the optimal closed-type barrier locations prioritized finding locations where debris-flow clusters merge or multi-impact (refer to Section 3.3.4.2). Such locations can effectively block multiple debris-flow clusters, reducing the overall cost (\$) by minimizing the number of barriers.

### 5.3.5 Results and discussion: optimal open-type barrier

A sensitivity analysis was performed on the SPEC-debris simulation results on the Mt. Umyeon Raemian and Sindonga region to determine the effect of optimization weighting factors and the open-type barrier performance on selecting optimal open-type barrier locations.

The optimization criteria weighting factors of Cases 4 to 6 (Table 5.4) were used. In Case 4, all factors were equal; hence, all were equally considered. For Case 5, the volume (V) and pressure (P) were prioritized so that the optimal barrier location would minimize the barrier size and reinforcement. For Case 6, the distance-from-road (D), number of closed-type barriers ( $N_c$ ), and number of open-type barriers ( $N_o$ ) were given priorities so that the optimal barrier location would maximize the accessibility and minimize the environmental impact of the barriers.

In addition, the ratio of the weighting factor between the number of closed-type barriers ( $N_c$ ) and the number of open-type barriers ( $N_o$ ) was maintained as 3:1 for Cases 5 and 6. A higher weighting factor was given to the closed-type barriers; therefore, the algorithm would prioritize placing open-type barriers over

the closed-type barriers. The closed-type barriers would be utilized only when the debris-flow disaster cannot be mitigated by using only the open-type barriers.

The optimal open-type barrier location selection procedure requires the engineer to select the following performance of the open-type barrier: speed ratio (SR) and trap ratio (TR), as defined in Eq. (3-14) and (3-15). Therefore, a sensitivity analysis determined the effect of open-type barrier performance on the optimal open-type barrier locations. Based on the range of speed ratio and trap ratio from the slit-type walls computed by Kim et al. (2020), a low performance and high-performance open-type barriers were tested:

Table 5.7 Open-type barrier performance for sensitivity analysis

Open-type barrier	speed ratio (SR)	trap ratio (TR)
low performance	0.6	0.6
high performance	0.4	0.4

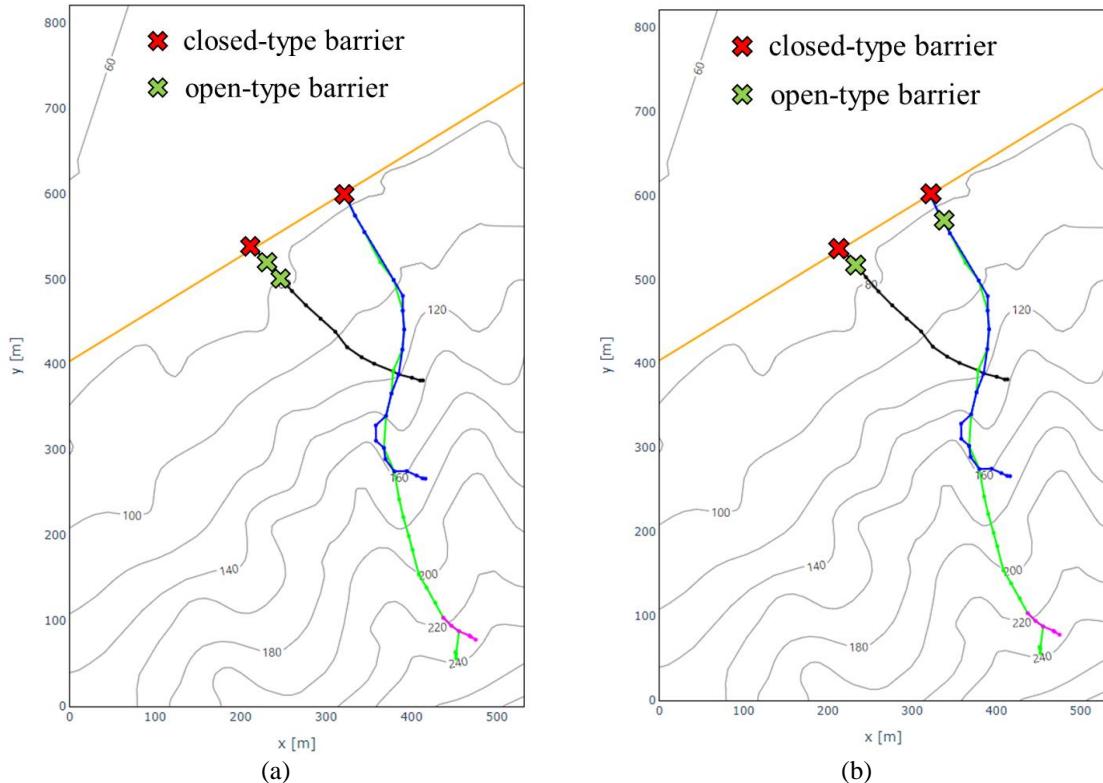
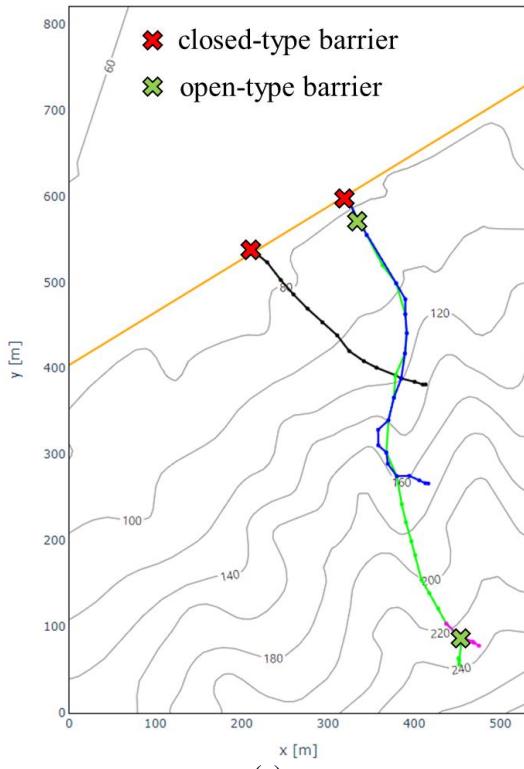
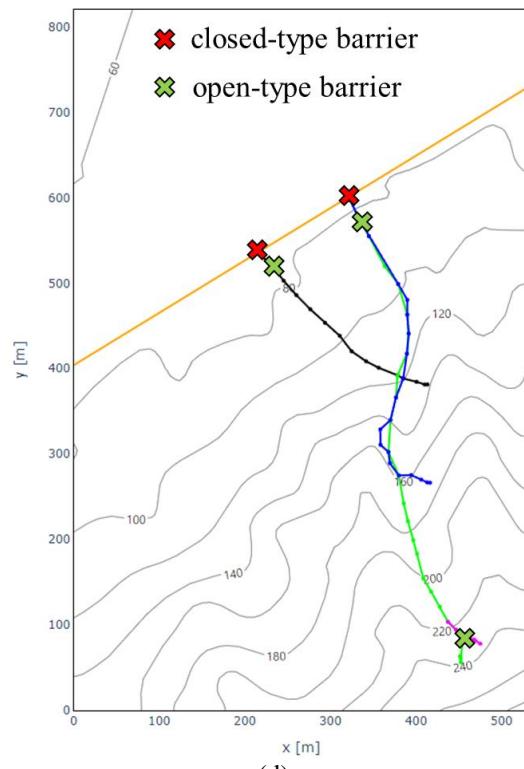


Figure 5.11 Optimal open and closed-type barrier locations based on open-type barrier performance and the optimization criteria at Mt. Umyeon Raemian region:

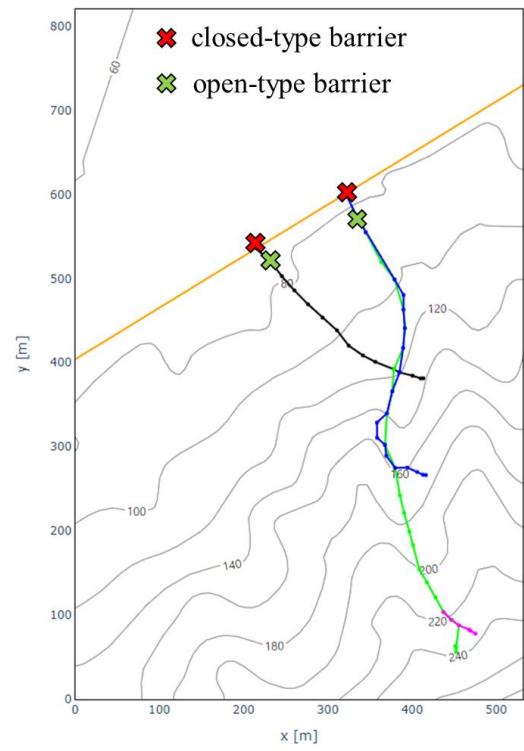
(a) case 4 – low performance and (b) case 4 – high performance



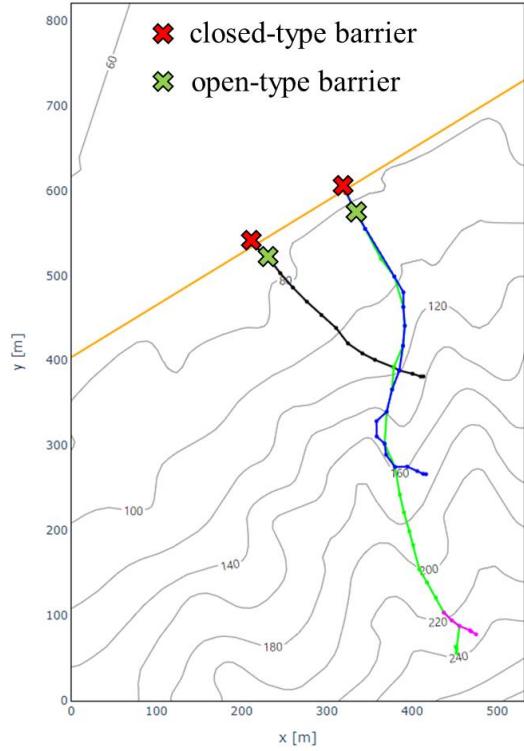
(c)



(d)



(e)



(f)

Figure 5.11 Optimal open and closed-type barrier locations based on open-type barrier performance and the optimization criteria at Mt. Umyeon Raemian region: (c) case 5 – low performance, (d) case 5 – high performance, (e) case 6 – low performance, and (f) case 6 – high performance

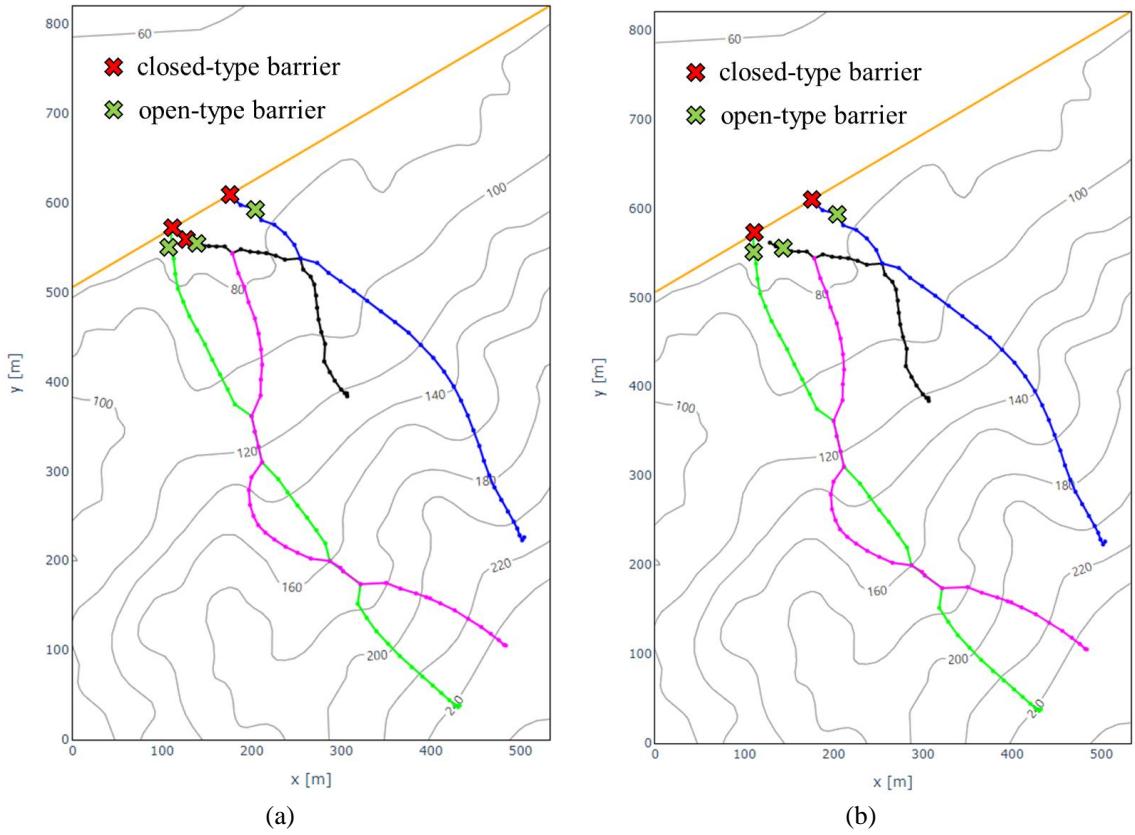


Figure 5.12 Optimal open and closed-type barrier locations based on open-type barrier performance and the optimization criteria at Mt. Umyeon Sindonga region: (a) low performance and (b) high performance

As shown in Figure 5.11, the sensitivity analysis shows that all six cases (Cases 4 to 6 and low/high barrier performance) selected similar optimal barrier locations in the Mt. Umyeon Raemian region. The notable difference is shown in Case 5, where the open-type barrier is placed nearer to the crest of the slope. The debris flow volume and pressure are smaller near the source location than debris flow at the toe of the slope.

The optimal barrier location procedure placed two closed-type barriers at the terminus, the boundary between the Mt. Umyeon and the residential area. The placement of closed-type barriers signifies that the optimization process could not find a feasible solution where successful mitigation is achieved by solely relying on open-type barriers. As described previously, the volume is massive due to entrainment, and the velocity is very rapid (28 m/s) at the toe of the slope. Therefore, the open-type barriers could not sufficiently reduce the volume and attenuate the velocity to an acceptable level of risk. Hence, the algorithm concluded that closed-type barriers would be more cost-effective than placing many open-type barriers.

The influence of open-type barrier performance can be seen in Cases 4 and 5. An open-type barrier is placed before the closed-type barrier at the terminus for high-performance open-type barriers. In these situations, the open-type barrier acts as a filter that reduces the volume and velocity of the on-coming debris-flow cluster; therefore, a smaller-sized closed-type barrier may be installed at the terminus. For low-performance open-type barriers, one of the closed-type barriers did not have open-type barriers placed before

the terminus. The difference can be explained using the cost-benefit ratio. For high-performance barriers, the benefit of a smaller closed-type barrier size outweighs the cost of constructing an open-type barrier. However, a low-performance open-type barrier did not sufficiently reduce the volume and velocity to justify the additional cost of installing an open-type barrier.

As shown in Figure 5.12, the sensitivity analysis shows that all six cases (Cases 4 to 6 and low/high barrier performance) selected similar optimal barrier locations in the Mt. Umeyeon Sindonga region. Similar to the Raemian region, the open-type barrier alone was insufficient to mitigate the debris-flow risk; hence, the model suggests the placement of closed-type barriers at the terminus locations. In these situations, the open-type barrier acts as a filter that reduces the volume and velocity of the on-coming debris-flow cluster; therefore, a smaller closed-type barrier would be sufficient. However, the result with a high-performance open-type barrier (Figure 5.12(b)) shows that the debris-flow risk can be mitigated with only open-type barriers in certain circumstances.

When both open-type and closed-type barriers are constructed for debris-flow mitigation, the open-type barriers are typically not placed very close to another closed-type barrier. Instead, the open-type barriers are placed regularly along the flowpath. Therefore, the weighting factors of distance-from-road (D) might need to be reduced, or additional criteria should be implemented to match typical open-type barrier design cases more closely.

#### 5.4 Conclusion

The accuracy and applicability of the developed SPEC-debris-barrier platform have been verified from the back-analysis of a small-scale flume laboratory test and the 2011 Mt. Umeyeon debris-flow disaster. In addition, the loss of kinetic energy due to collision in the SPEC-debris model has been successfully demonstrated. Additional case studies are desirable to test the developed platform further but will require a considerable expansion of case study and laboratory test database; hence, further verification will be recommended for future research.

## Chapter 6. Conclusions and Recommendations

### 6.1 Conclusion and summary

The thesis has contributed to debris-flow hazard assessment and barrier mitigation against debris-flow disasters by developing the SPEC-debris-barrier platform. The developed platform is an all-in-one computational tool for analyzing the followings: debris-flow hazard assessment, optimal barrier design, and barrier performance. The highlights of the work presented in Chapters 1 to 5 follow:

- The need for efficient methods to design cost-effective optimal barriers, especially considering both closed and open-type barriers, was addressed as a mitigation measure against debris-flow hazards for regions nearby debris-flow susceptible locations. The need for core technology to efficiently analyze debris-flow propagation, design optimal barriers, and evaluate barrier performance was identified.
- A review of existing runout simulation methods for hazard assessment and evaluating barrier performance was presented. The inherent limitation of the existing empirical and depth-averaged continuum model for barrier performance evaluation was identified due to not incorporating the energy loss due to collision. In addition, the computational inefficiency and hardware required to perform a full 3D continuum or discontinuum simulation for barrier performance evaluation were addressed. Therefore, a need for a depth-averaged continuum model incorporating the effect of barrier collision for evaluating barrier performance was identified.
- The SPEC-debris runout simulation method was developed to perform a runout analysis and evaluate the barrier performance. The model utilized the “equivalent fluid” (Hung, 1995) principle and shallow water equation (SWE) or St. Venant equations. The runout mass is discretized into particles. The following governing equations are utilized to compute the propagation of the runout mass: smoothed particle hydrodynamics (SPH), pathway algorithm, energy conservation model, and collision model.
- The SPH interpolation was adopted to compute the depth and depth gradient of incompressible, depth-averaged runout material using the Gaussian kernel function. The smart neighboring particle search method was utilized to account for the separation of the fluid particle by a barrier/wall. The boundary ghost particles were utilized to accurately interpolate the fluid near the digital elevation model (DEM) boundary.
- The pathway model adopts the SWE to compute the direction of propagation based on the following factors: gravity, hydrodynamic pressure gradient, basal rheological resistance, momentum transfer from entrainment phenomena, and inertia.
- For particles traveling along the surface of DEM, the leapfrog method, a second-order numerical procedure for solving explicit simulations, was used to compute the following particle location. However, particle colliding against the barrier or in an airborne state is assumed to be in parabolic motion, affecting the gravitational acceleration.

- The energy conservation model computes the speed of particles that travels on the ground and are airborne. The model balances the following forms of energy: kinetic energy, gravitational potential energy, pressure gradient potential energy, and energy loss due to friction.
- The SPEC-debris method distinguishes itself from other depth-averaged runout models by incorporating loss of kinetic energy from collisions. The SPEC-debris utilizes both continuous and discrete collision detection to identify collisions. The velocity change after a collision is governed by the conservation of linear momentum and the coefficient of restitution (COR) coefficients.
- A review of the existing barrier design frameworks utilizing empirical or numerical analysis was presented. The applicability of barrier design with an empirical analysis approach is limited due to the inherent limitation of empirical analysis for identifying debris-flow propagation routes, i.e., flowpaths, simplistic barrier performance evaluation, and heavy reliance on engineering judgment. The barrier design with a numerical analysis approach is relatively slow and demanding on the hardware; hence, it heavily trades the optimization procedure with computational efficiency. Therefore, an optimal barrier design framework was needed to utilize the optimization algorithm to search for an optimal barrier design for closed and open-type barrier designs.
- The debris-flow cluster analysis groups the debris-flow particles from the SPEC-debris simulation into debris-flow clusters. Tracking the propagation and characteristics (velocity, depth, volume, and pressure) of the debris-flow clusters generates an overview of the entire debris-flow propagation.
- The debris-flow cluster analysis results are stored in a debris-flow network data structure. The network data structure is composed of nodes representing the characteristics of the debris-flow clusters at a particular location, i.e., debris-flow cluster location (DFL) and directional link representing the flow of debris from one DFL to another. Connecting the directional link traces a flowpath. The nodes are classified by the interaction between debris-flow clusters at the DFL.
- The network data structure can quickly evaluate whether the set of barriers (both closed and open-type barriers) has successfully mitigated the risk of infrastructure damage due to debris-flow hazards.
- The stochastic tabu search optimization algorithm utilizes the debris-flow network data structure to search for the optimal barrier (both closed and open-type barrier) locations that satisfy the following design criteria: number of barriers, volume, pressure, and distance-from-road. The optimality of a set of barrier locations is evaluated through the cost analysis that accounts for the relative importance of design criteria, i.e., optimization weighting factors.
- The developed SPEC-debris simulation method and optimal barrier design framework were incorporated into the SPEC-debris-barrier platform.
- Using the SPEC-debris-barrier platform, the SPEC-debris method for evaluating barrier performance and debris-flow runout simulation was verified through a back-analysis of a small-scale flume experiment and case study of the 2011 debris-flow incident at Mt. Umyeon.

- Using the SPEC-debris-barrier platform, the applicability of the developed optimal barrier design method was demonstrated in the case study of the 2011 debris-flow incident at Mt. Umeyon. The sensitivity analysis of the influence of optimization criteria weighting factors on the closed and open-type barrier locations was explored. Finally, the influence of open-type barrier performance on open-type barrier locations was also presented.

## 6.2 Recommendation for future work

The SPEC-debris-barrier platform and its analysis tools are considered fit for hazard assessment and barrier designs; however, several limitations have been identified, which could be pursued for future research. The recommended future work consists of the following:

- The limitation for forward analysis on the barrier performance is the difficulty in determining the coefficient of restitution (COR) value between the fluid particle and barrier walls. Although predicting the COR of impacting elastic-perfectly plastic sphere has been researched (Jackson et al., 2010), the COR between the debris-flow and barrier is limited. Therefore, investigation through laboratory experiments or back-analysis to determine the COR would be recommended for future work.
- The validation of the SPEC-debris simulation method should be continued as more experimental and case studies are available.
- The influence of other popularly-used smoothed particle hydrodynamics (SPH) kernel functions, such as Wendland (Wendland, 1995) and cubic kernels, on the depth and depth gradient computation should be investigated. These SPH kernel functions save computational effort by limiting the SPH kernel influence to be within  $2l$ , where  $l$  is the smoothing length.
- The hydrodynamic pressure equation determines the impact pressure with an empirical model based on the Froude number. The SPH interpolation method for computing the impact pressure from the collision can be investigated.
- The current simplistic implementation of location-based material properties assignment is challenging to model the change of rheological parameters. For example, a function-based method that can modify the material properties would provide the following advanced runout simulation capabilities: shear-thickening, reactivation of slow-moving landslides, and two-phase flow.
- The optimization criteria weighting factors in the model are based on engineering judgment; however, such weighting factors may be more rigorously and objectively derived from performing cost analysis on barrier construction.
- The optimal open-type barrier location selection model is too focused on minimizing the number of barriers and placing the barriers closer an access route, which minimizes distance-from-road (D). A different optimal open-type barrier location selection model could be developed that optimizes the effect arrangement and spacing of open-type barriers along the flowpath.

- A comprehensive parametric study should be performed to analyze the effect of other input variables described in Chapter 4 and Appendix A to analyze their influence on the SPEC-debris runout simulation and optimal barrier design framework. Especially, the *particle number per cell* variable, which determines the fluid particle number and radius, requires further investigation. Finally, the influence of particle size in the barrier performance analysis should be explored.
- The current version of the SPEC-debris-barrier platform requires the user to input the initial landslide locations. These locations are often determined by performing a combination of rainfall infiltration transient seepage and slope stability analysis. Therefore, the development of the platform incorporating the infiltration and slope stability analysis models can be pursued in the future version.
- The SPEC-debris-barrier platform implemented multi-core parallel processing on the CPU; however, the acceleration of computation process from utilizing GPU process can be developed in the future.
- The application of the SPEC-debris-barrier platform for analyzing other runout hazards, such as flash flooding and snow avalanche, and determining optimal barrier locations as mitigation against those runout hazards could be investigated.

## Appendix A. SPEC-debris-barrier Platform JSON

### A.1 Input JSON

The general template for an input JSON file is shown in Figure 4.9. The user must select one of the valid analysis keywords and assign appropriate data to the input variables. The valid analysis keywords and variable names are summarized in Tables 4.1 and A.1.

The user should follow the JSON file syntax (refer to Table A.2) when writing the input JSON file. It should be noted that JSON is case-sensitive, which means capital and lowercase letters are differentiated; therefore, the user should be careful when using a capital letter. The analysis keywords and variable names are case-sensitive; hence, a capital letter may cause an error.

Table A.1 Input categories and valid variable names

Input Category (Section No.)	Variable Names
Basic Information (A.2)	<i>project name, folder path</i>
DEM, Roads, and Structures (A.3)	<i>flowpath file name, source file name, material file name, grid x array, grid y array, grid size, flowpath z array, material id array, road xy, wall info</i>
Initial and Material Properties (A.4)	<i>initial u_x, initial u_y, material, overall_fb_0 coefficient of restitution (COR), initial SPH</i>
Algorithm Option (A.5)	<i>interpolation method, entrainment model, cluster boundary method</i>
Control Parameter (A.6)	<i>time step interval, maximum simulation time, particle number per cell, gravitational acceleration, free fall angle variation, local cell sizes, l_dp_min, merge overlap ratio, concave hull algorithm parameter, max cpu num</i>
Optimal Barrier Location Selection Parameter (A.7)	<i>optimal barrier location selection option, closed barrier optimization criteria, combined barrier optimization criteria, open barrier performance, critical vulnerability index (VI), reinforced concrete (RC) wall</i>
Plot Option (A.8)	<i>static plot option, animation option, max parameter legend, marker size, line width, layout width, layout height</i>
Output Option (A.9)	<i>csv output, output optimal step, open plot, decimal points</i>
SPEC-debris Cluster (A.10)	<i>SPEC_debris analysis cluster data</i>

Table A.2 JSON data type and syntax

JSON Data Types	Syntax	Example
string	<ul style="list-style-type: none"> <li>• a sequence of characters (number, letter, etc.), which is recognized as a text in JSON</li> <li>• must be written in double quotes (" ")</li> <li>• case sensitive (differentiating capital and lowercase letters)</li> </ul>	"density"
integer	<ul style="list-style-type: none"> <li>• mathematical integer data</li> <li>• no decimal point (.)</li> </ul>	1000
float	<ul style="list-style-type: none"> <li>• mathematical real number data</li> <li>• non-zero values after the decimal point (.)</li> </ul>	3.14
boolean	<ul style="list-style-type: none"> <li>• either true or false state</li> <li>• all lowercased</li> </ul>	true, false
null	<ul style="list-style-type: none"> <li>• a null value, or no value at all</li> <li>• it is not the same as 0, False, or an empty string</li> </ul>	null
array	<ul style="list-style-type: none"> <li>• a collection of elements in a sequence</li> <li>• must be enclosed by squared brackets ([ ])</li> <li>• elements are separated with a comma (,)</li> <li>• the element can be any valid JSON data type</li> <li>• the order is very important</li> </ul>	[3, "pi"]
object	<ul style="list-style-type: none"> <li>• a collection of sets of key and value pairs</li> <li>• must be enclosed by curly brackets ({ })</li> <li>• the key and value are separated by a colon (:)</li> <li>• the key must be a string data type</li> <li>• value can be any valid JSON data type</li> <li>• a comma (,) separates each key and value pairs</li> <li>• the order is not important</li> </ul>	{"pi": 3.14, "h": 180}

An appropriate data type and valid layout must be used when assigning data to the input variables. In Appendix A, the input variables will be described in detail, and an appropriate data format will be specified.

## A.2 Basic information

The basic information category specifies the naming of the platform outputs and the saving folder location for the output through the following variables:

- *project name*
  - Description: name of the project and form the basic template for naming the output files
  - Data format: string
  - Example: "sample\_SPECDB\_pro1"
  - Notes: Use the underscore symbol ( \_) instead of a space ( )
  - Analysis usage: *SPEC-debris*, *SPEC-debris-wall*, *SPEC-debris-closed*, *SPEC-debris-combined*, *closed*, *combined*

- *folder path*
  - Description: folder location of the input topography files and folder to save the output files
  - Data format: string
  - Example: “C:/Users/User/Document/pro1”
  - Notes:
    - The forward-slash (/) symbol is used to separate the folder names
    - The forward slash is not required at the end of the *folder path*
    - Recommended to specify computer drive folders (e.g., “C:/” or “D:/”) in the *folder path*
  - Analysis usage: *SPEC-debris*, *SPEC-debris-wall*, *SPEC-debris-closed*, *SPEC-debris-combined*, *closed*, *combined*

## A.3 DEM, road, and structures

The DEM, road, and structures input category specifies the spatial data required for the SPEC-debris-barrier platform, such as the mountain topography, landslide locations, material locations, transportation routes, barriers installation, and building locations.

### A.3.1 Topography, source, and material

The SPEC-debris-barrier platform can import external files, such as csv text and LiDAR, containing spatial data for defining topography, initial sliding mass locations, and material distributions. The input JSON file requires the user to provide the filenames of the external files using the following variables:

- *flowpath file name*
  - Description: filename of the flowpath digital elevation model (DEM) containing the topography data
  - Data format: string
  - Example: “mountain\_DEM.csv”
  - Notes:
    - Requires the full filename with the extensions
    - Use the underscore symbol ( \_) instead of space ( )
    - Accepts the following three file types:
      - csv file type (“.csv”) is comma-separated text format with XYZ dataset
      - grd file type (“.grd”) is the Golden Software Surfer 6 ASCII Grid file format with elevation (Z) data
      - asc file type (“.asc”) is the ArcGIS Esri ASCII file format with elevation (Z) data
      - las file type (“.las”) is LiDAR data with elevation (Z) data
  - Analysis usage: *SPEC-debris*, *SPEC-debris-wall*, *SPEC-debris-closed*, *SPEC-debris-combined*, *closed*, *combined*

- *source file name*
  - Description: filename for the data containing sliding mass location at initial (time step = 0). Instead of the elevation like DEM, the source file provides a particular XY location's vertical depth of sliding mass. Initial sliding mass locations are defined as locations with a non-zero value of sliding mass depth (h)
  - Data format: string
  - Example: “case1\_source.csv”
  - Notes:
    - Must have the same X and Y grid coordinates as the flowpath DEM from the *flowpath file name*
    - Requires the full filename with the extensions
    - Use the underscore symbol ( \_) instead of space ( )
    - Accepts the following three file types:
      - csv file type (“.csv”) is comma-separated text format with XYh dataset
      - grd file type (“.grd”) is the Golden Software Surfer 6 ASCII Grid file format with depth (h) data
      - asc file type (“.asc”) is the ArcGIS Esri ASCII file format with depth (h) data
      - las file type (“.las”) is LiDAR data with depth (h) data
  - Analysis usage: *SPEC-debris*, *SPEC-debris-wall*, *SPEC-debris-closed*, *SPEC-debris-combined*
- *material file name*
  - Description: filename for the data containing the distribution of material properties. Instead of the elevation like DEM, the material file provides the material ID number at a particular XY location. If the number of material properties is one (1), the *material file name* is optional
  - Data format: string or null
  - Example: “materialID\_location.csv”
  - Notes:
    - If specified, it must have the same X and Y grid coordinates as the flowpath DEM from the *flowpath file name*
    - The material ID should start from one (1)
    - If the number of material properties is one (1), the *material file name* can be defined as JSON null data type (e.g., “*material file name*”: null)
    - Requires the full filename with the extensions
    - Use the underscore symbol ( \_) instead of space ( )
    - Accepts the following three file types:
      - csv file type (“.csv”) is comma-separated text format with XY-material ID dataset
      - grd file type (“.grd”) is the Golden Software Surfer 6 ASCII Grid file format with material ID data

- asc file type (“.asc”) is the ArcGIS Esri ASCII file format with material ID data
- las file type (“.las”) is LiDAR data with material ID data
- Analysis usage: *SPEC-debris*, *SPEC-debris-wall*, *SPEC-debris-closed*, *SPEC-debris-combined*

The flowpath, source, and material data imported from the external files described by the previous three variables (*flowpath file name*, *source file name*, and *material file name*) are transformed into an array format used in the SPEC-debris simulation. The following variables can be used to load the array format of the flowpath and material data for the optimal barrier location selection without performing the SPEC-debris analysis:

- *grid x array* and *grid y array*
  - Description: defines the array containing grid X (for *grid x array*) and Y (for *grid y array*) coordinates of the digital elevation model (DEM) data
  - Data format: 1D array with float
  - Example: [0.0, 1.0, … 100.0]
  - Notes: the DEM data is assumed to be a square grid
  - Analysis usage: *closed*, *combined*
- *grid size*
  - Description: defines the grid spacing between the grid X and Y coordinates of the digital elevation model (DEM) data
  - Data format: float
  - Example: 0.5
  - Notes: the DEM data is assumed to be a square grid; hence, the grid spacing between the X and Y coordinates are equal, i.e., *grid size* = DX = DY
  - Analysis usage: *closed*, *combined*
- *flowpath z array*
  - Description: defines the cell elevation of the digital elevation model (DEM) data
  - Data format: 2D array with float
  - Example: [[0.0, 0.0, … 0.5],  
[0.0, 0.2, … 0.5],  
…  
[100.0, 102.0, … 120.5]]
  - Notes: the row and the column of the array refer to the Y and X grid position, respectively
  - Analysis usage: *closed*, *combined*
- *material id array*
  - Description: defines the material ID assigned for the cell
  - Data format: 2D array with integer

- Example:  $[[1, 1, \dots, 1],$   
 $[1, 2, \dots, 1],$   
 $\dots$   
 $[1, 2, \dots, 2]]$

- Notes:

- o The row and the column of the array refer to the Y and X grid position, respectively

- Analysis usage: *closed, combined*

In general, the user need not prepare the data for these variables. The data for these variables are automatically generated when either the *SPEC-debris-closed* or the *SPEC-debris-combined* keyword is selected. Therefore, the user can focus on performing multiple optimal barrier location selection analyses using the same SPEC-debris analysis results.

### A.3.2 Road

As explained in Chapter 3, the distance-from-road (D) parameter is one of the key optimization criteria for selecting optimal barrier locations. The location of the transportation route near the base of the mountain is defined by a linear line between two sets of XY coordinates, as shown in Figure 3.10. The user can assign these two XY points along the road:

- *road xy*
  - Description: define a linear transportation route, i.e., road, segment with two XY points
  - Data format: 2D array with float –  $[[X_1, Y_1], [X_2, Y_2]]$
  - Example:  $[[0.0, 100.0], [100.0, 150.0]]$
  - Notes:
    - o If specified, it must have the same X and Y grid coordinates as the flowpath DEM from the *flowpath file name*
    - o The material ID should start from one (1)
  - Analysis usage: *SPEC-debris, SPEC-debris-wall, SPEC-debris-closed, SPEC-debris-combined, closed, combined*

### A.3.3 Structures

The SPEC-debris-barrier platform allows a wide range of barriers and buildings to be modeled for SPEC-debris analysis. The platform assumes that the wall of a barrier or building is solid without any opening; therefore, the platform is most suited to modeling closed-type barriers, slit-type barriers, and baffles.

The platform models the structure by modifying the elevation of the flowpath according to four kinds of elevation modification methods. The modified flowpath DEM data simulates the collision (see Section 2.2.5). The flowpath DEM is modified by the following procedure, as shown in Figure A.1:

- (1) define the boundary of a structure (red shape in Figure A.1),
- (2) find all the XY grid that lies within the structure boundary (green points in Figure A.1),

(3) modify the elevation of all XY grids within the structure boundary based on the elevation modification options, as shown in Figure A.1 (c)-(f):

1.  $Z_{in}$ : all elevations are modified to be equal to the specified Z value
2.  $\Delta Z$ : all elevations are changed by adding  $\Delta Z$  value to the original flowpath elevation
3. Min  $\Delta Z$ : all elevations are modified by adding  $\Delta Z$  to the minimum elevation of the original flowpath DEM
4. Max  $\Delta Z$ : all elevations are modified by adding  $\Delta Z$  to the maximum elevation of the original flowpath DEM

On top of modifying the flowpath DEM, the boundary of the structure is stored separately. The boundary data distinguishes the flowpath from the structure location, which is impossible in existing depth-averaged runout models where only flowpath DEM can be modeled.

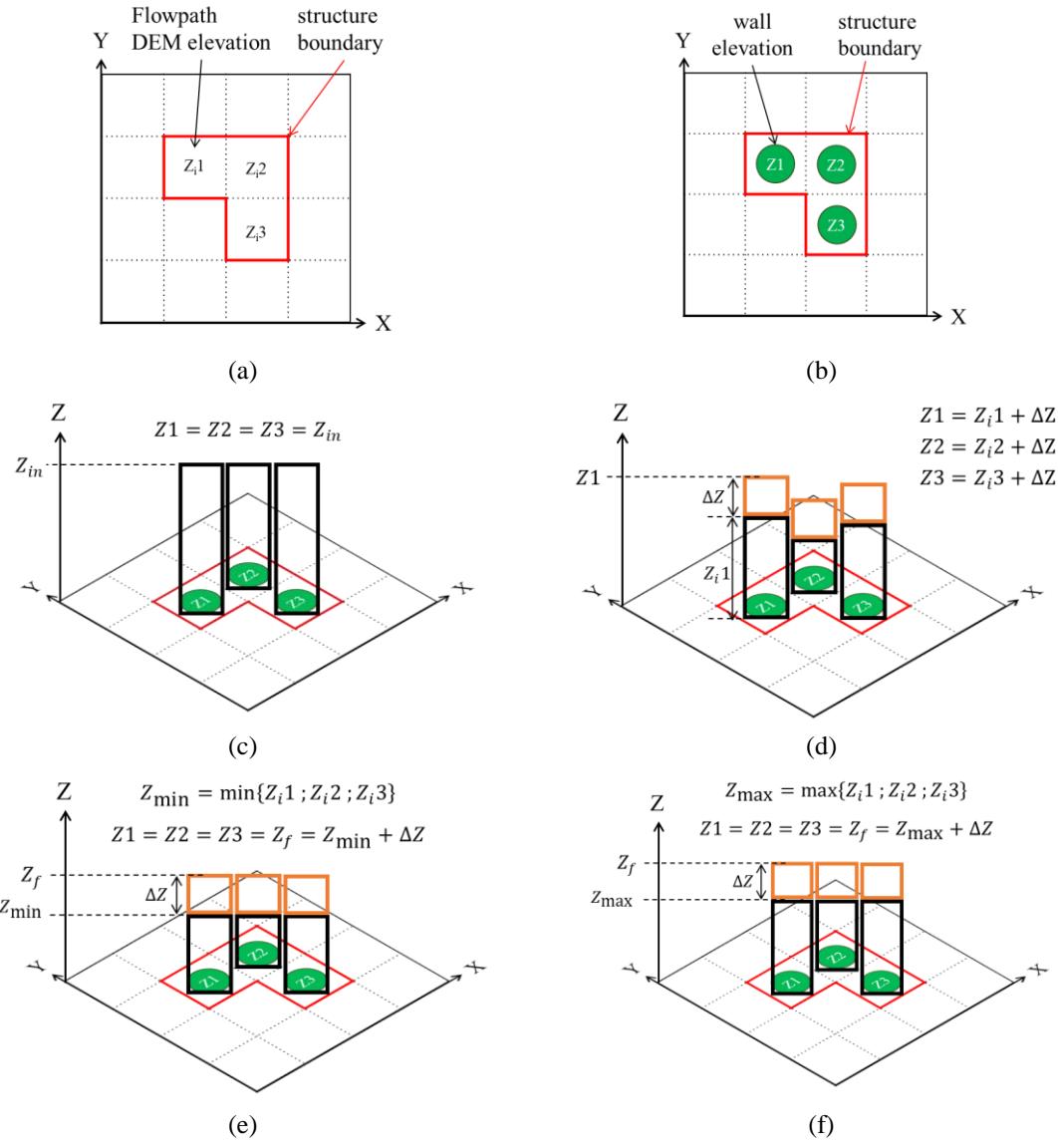


Figure A.1 Flowpath DEM elevation modification procedure: (a) structure boundary, (b) modification of elevation inside structure boundary, (c)  $Z_{in}$  modification, (d)  $\Delta Z$  modification, (e) min  $\Delta Z$  modification, and (f) max  $\Delta Z$  modification

The SPEC-debris-barrier platform can model three types of barriers and one generalized building structure, i.e., four types of structures. Any closed or open-type barrier with a quadrilateral wall shape can be modeled with “P” and “V” structure types referring to parallel and V-shape arrangements of walls, respectively (Figure A.2(a)-(d)). The “C” (circular) structure type can be used to model any open-type barrier with cylinder columns or circular piles (Figure A.2(e)). Lastly, the “BD” (building) structure type can define any structure shape (Figure A.2(f)); therefore, it is helpful for modeling buildings and barriers that cannot be modeled with the other three structure types.

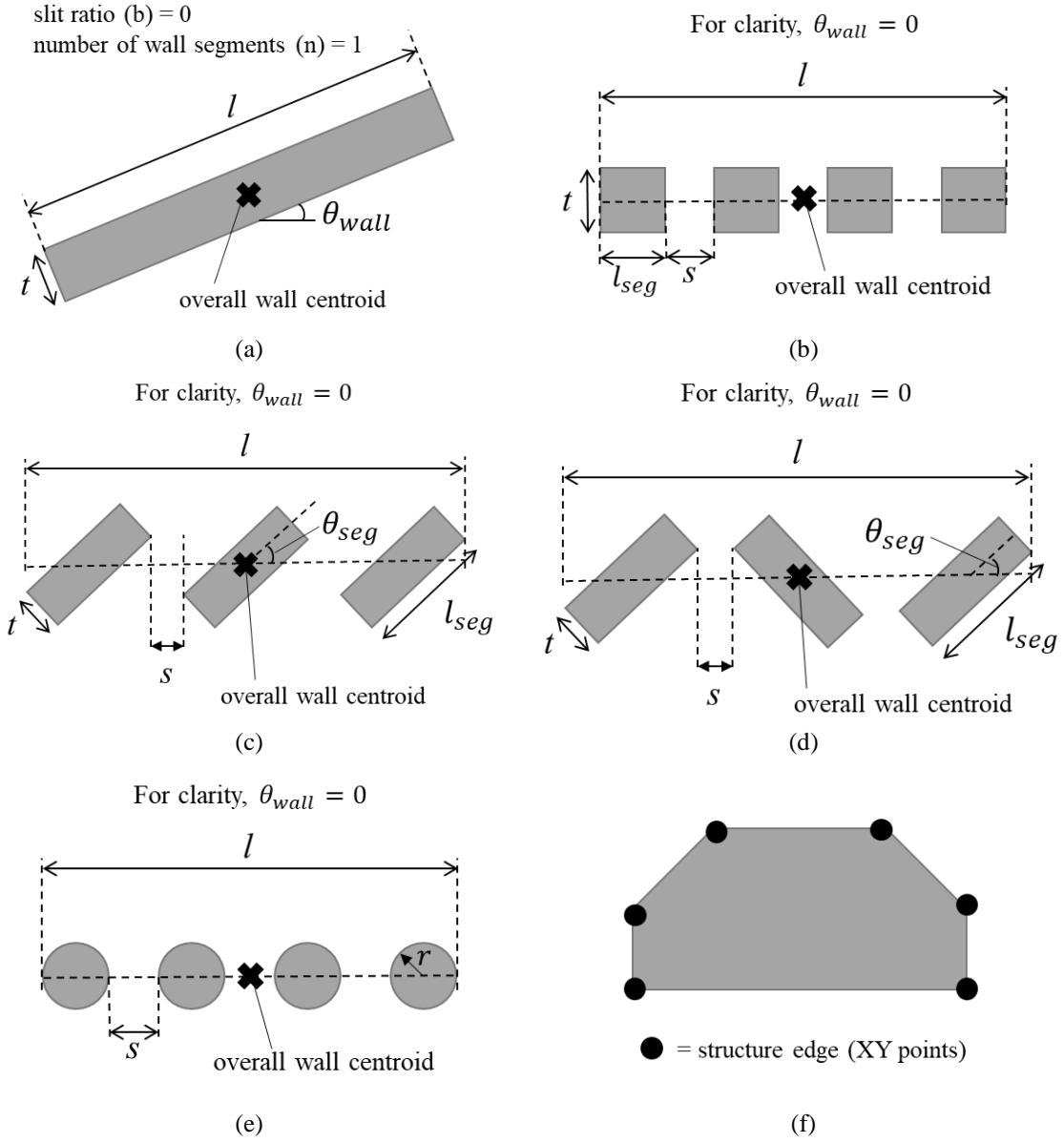


Figure A.2 Barrier and design parameters: (a) closed-type barriers (“P”), (b) box-shaped baffles (“P”), (c) parallel slit-type barriers (“P”), (d) V-shape arranged slit-type barriers (“V”), (e) cylinder baffles (“C”), and (f) general building (“BD”)

For open-type barriers, the slit ratio ( $b$ ) is defined by Eq. (A-1), assuming that the length of the wall or baffle is the same and the spacing between the wall or baffle is equal:

$$b = \sum(s)/l = s(n - 1)/l \quad (\text{A-1})$$

where  $s$  is spacings between wall segments or baffles,  $n$  is the total number of wall segments or baffles, and  $l$  is the total wall length. The slit ratio  $b$  value must be smaller than one (1). The length of a wall segment or box baffle is computed by Eq. (A-2):

$$l_{seg} = l(1 - b)/n \quad (\text{A-2})$$

where  $l_{seg}$  is the wall segments or baffles length,  $l$  is the total wall length,  $b$  is the slit ratio, and  $n$  is the total number of wall segments or baffles.

The wall information is provided through the variable *wall info*:

- *wall info*
  - Description:
    - Defines the structure design parameters (Figure A.2)
    - For “P” and “V” structure types, the *wall info* object needs to define:
      - “wall type”: (string) either “P” for parallel orientation or “V” for an alternative orientation
      - “slit ratio”: (float) between 0 and 1
      - “number of wall segments”: (integer) 1 or higher
      - “orientation of wall segments (Polar)": (float) overall wall orientation relative to the East
      - “orientation of wall overall (Polar)": (float) wall segment orientation relative to the overall wall orientation
      - “wall thickness”: (float) thickness of wall segments
      - “wall length”: (float) overall length of the whole wall section
      - “wall elevation option”: (integer) 1, 2, 3, or 4
      - “elevation of height”: (float) specified elevation  $Z_{in}$  or height  $\Delta Z$
      - “wall centroid X”: (float) centroid X coordinate of the whole wall section
      - “wall centroid Y”: (float) centroid Y coordinate of the whole wall section
  - For “C” structure types, the *wall info* object needs to define:
    - “wall type”: (string) “C” for circular structures
    - “slit ratio”: (float) between 0 and 1
    - “number of cylinder segments”: (integer) higher than 1
    - “orientation of wall overall (Polar)": (float) overall wall orientation relative to the East
    - “cylinder radius”: (float) half the thickness or the radius of the cylinder baffle
    - “wall length”: (float) overall length of the whole wall section
    - “wall elevation option”: (integer) 1, 2, 3, or 4
    - “elevation of height”: (float) specified elevation  $Z_{in}$  or height  $\Delta Z$
    - “wall centroid X”: (float) centroid X coordinate of the whole wall section

“wall centroid Y”: (float) centroid Y coordinate of the whole wall section

- For “BD” structure types, the *wall info* object needs to define:

“wall type”: (string) “BD” for building structure

“wall elevation option”: (integer) 1, 2, 3, or 4

“elevation of height”: (float) specified elevation  $Z_{in}$  or height  $\Delta Z$

“wall XY points”: (array – float) input XY coordinates of the edges of the building in the plan view, and is written as [[X1, Y1], [X2, Y2], … [Xn, Yn]]

- Data format: object or null

- Example: for a closed-type barrier (see Figure A.2(a))

“*wall info*”:

```
{“wall type”: “P”, “slit ratio”: 0, “number of wall segments”: 1,  
“orientation of wall segments (Polar)": 30.0, “orientation of wall overall (Polar)": 0.0,  
“wall thickness": 1.0, “wall length": 10.0,  
“wall elevation option": 1, “elevation of height": 30.0,  
“wall centroid X": 100.0, “wall centroid Y": 50.0}
```

- Notes:

- If no wall is modeled, null is assigned to *wall info*
- The angles are in a polar coordinate system where the East is the  $0^\circ$
- The angles are limited from  $-90^\circ$  to  $90^\circ$
- For “P”, “V”, and “C” structure types, the “orientation of wall overall (Polar)” is relative to the positive X direction, or the East
- For “P” and “V” structure types, the “orientation of wall segments (Polar)” is relative to the direction of the overall wall, as specified at “orientation of wall overall (Polar)”
- For “P” and “V” structure types, a closed-type barrier is modeled if the “slit ratio” parameter equals one (1) or the “number of wall segments” parameter equals zero (0)
- The “wall elevation option” option refers to the following DEM elevation modification:
  - 1 =  $Z_{in}$  (Figure A.1(c))
  - 2 =  $\Delta Z$  (Figure A.1(d))
  - 3 = min  $\Delta Z$  (Figure A.1(e))
  - 4 = max  $\Delta Z$  (Figure A.1(f))

- Analysis usage: *SPEC-debris-wall*

## A.4 Initial and material properties

The initial and material properties input category specifies the initial conditions of the debris-flow, geotechnical properties, and erosion conditions for the SPEC-debris-barrier platform.

#### A.4.1 Initial velocity

The SPEC-debris-barrier platform allows the user to specify the initial velocity of debris-flow. For example, if the runout simulation starts at landslide failure, the initial velocity is assumed to be zero (0). However, suppose the user wants to simulate runout with different initial conditions, such as flume laboratory tests. In that case, the initial velocity can be specified in the input JSON file using the following variables:

- *initial u\_x* and *initial u\_y*
  - Description: velocity of the debris-flow particles in X and Y directions at initial (time step = 0)
  - Data format: float or null
  - Example: “*initial u\_x*”: 1.0, “*initial u\_y*”: null
  - Notes:
    - If the initial velocity magnitude is zero (0), then the user can either specify a zero (0) value or leave it as a null data type
    - The initial velocity can be expressed as negative numbers to assign the direction. The platform uses the East and the North as the positive X and Y directions, respectively
    - If specified, the initial velocity is applied equally to very particles
  - Analysis usage: *SPEC-debris*, *SPEC-debris-wall*, *SPEC-debris-closed*, *SPEC-debris-combined*

#### A.4.2 Rheology, internal friction, and entrainment

The user needs to specify the geotechnical properties of the debris-flow, such as density  $\rho$ , Voellmy rheological parameters  $f$ , internal friction angle  $\phi'$ , and entrainment growth rate  $E_s$ . The details of these material properties are described in Sections 2.2.3.3 and 2.2.4.1. The material properties are specified through:

- *material*
  - Description:
    - Set of material properties assigned to each material ID number
    - For each material ID, the following must be defined:
      - “*f*”: (array – float) basal frictional coefficient ( $f_b$ ) and turbulence coefficient ( $f_T$ ) of the Voellmy rheological model, and is written as [ $f_b, f_T$ ]
      - “*phi*”: (float) internal frictional angle ( $\phi'$ ) for lateral earth pressure coefficients
      - “*Es*”: (float) entrainment growth factor
      - “*density*”: (float) density of the material
      - “*max erode depth*”: (float) maximum erodible depth
    - Data format: object

- Example: material properties for material ID number 1

```

“material”: {
    “1”: {
        “f”: [0.1, 0.005],
        “phi”: 30.0,
        “Es”: 0.002,
        “density”: 2000.0,
        “max erode depth”: 4.0
    }
}

```

- Notes:

- o The material ID number must be written as an integer-string, i.e., an integer number between double quotation marks (“ ”)
- o If the eroding depth at a particular cell location exceeds the “*max erode depth*”, then no entrainment occurs, and the “*Es*” value becomes zero (0)
- Analysis usage: *SPEC-debris*, *SPEC-debris-wall*, *SPEC-debris-closed*, *SPEC-debris-combined*, *combined*

The optimal open-type barrier location selection uses a simplified energy conservation model with Coulomb frictional rheology to compute the propagation of debris-flow after colliding with an open-type barrier, as described in Section 3.3.4.4. The Coulomb frictional coefficient ( $f_{b0}$ ) can be computed from Voellmy rheology by Eq. (A-3):

$$f_{b0} = f_b + f_T \left( \frac{\bar{u}^2}{\bar{h}} \right) \quad (\text{A-3})$$

where  $f_b$  is the Voellmy basal frictional coefficient,  $f_T$  is the Voellmy turbulence coefficient,  $\bar{u}$  is the average velocity magnitude of debris-flow, and  $\bar{h}$  is the average depth of debris-flow. Otherwise, the user can prescribe the Coulomb frictional coefficient ( $f_{b0}$ ) through:

- *overall\_fb\_0*
  - Description: Coulomb rheology frictional coefficient
  - Data format: float or null
  - Example: “*overall\_fb\_0*”: 0.15
  - Notes: the Coulomb frictional coefficient ( $f_{b0}$ ) is only used for combined optimal barrier location selection; hence, it is assigned as a null data type
  - Analysis usage: *combined*

### A.4.3 Coefficient of restitution (COR)

The SPEC-debris analysis uses the coefficient of restitution (COR) values to compute the momentum transfer of debris-flow particles after collisions. The COR values are described in detail in Section 2.2.5. The COR for inter-particle collision and particle-wall collision can be assigned through:

- *coefficient of restitution (COR)*
  - Description: coefficient of restitution for inter-particles collision (COR\_p2p) and for particle-wall collision (COR\_p2w)
    - The following must be defined:
      - “*particle with particle*”: (float) COR\_p2p value
      - “*particle with wall*”: (float) COR\_p2w value
    - Data format: object
    - Example:

```
“coefficient of restitution (COR)": {  
    “particle with particle”: 0.7,  
    “particle with wall”: 0.6  
}
```
    - Notes: the COR value must lie from 0 to 1
    - Analysis usage: *SPEC-debris-wall*

### A.4.4 Initial SPH interpolation

The SPEC-debris analysis allows the user to select the option for interpolating the depth of the initial runout mass, i.e., source, through SPH interpolation. The initial SPH interpolation would generate depth distribution similar to a spherical slope failure surface; therefore, the initial depth near the edges of the source will be taper off, and the largest failure depth will be located at the center of the source. The SPH depth interpolation at the initial (time step = 0) would more accurately model the actual landslide failure surface for an unconfined runout source, such as a shallow landslide.

However, the SPEC-debris-barrier platform allows users to disable the initial SPH interpolation of depth at time step = 0. For confined runout mass, such as dam-break or flume test, the depth of the runout mass is defined by the container; hence, the depth should not be tapering off near the edges of the fluid body. Therefore, the depth value is taken directly from the input source file without initial interpolation.

The user can specify the *initial SPH* variable to enable or disable initial depth interpolation through:

- *Initial SPH*
  - Description: option to enable initial depth interpolation with SPH interpolation
  - Data format: boolean
  - Example: true
  - Notes: the default value is true
  - Analysis usage: *SPEC-debris-closed*, *SPEC-debris-combined*, *SPEC-debris*, *SPEC-debris-wall*

## A.5 Algorithm options

The algorithm option specifies the interpolation methodology, entrainment model, and algorithm to determine the debris-flow cluster boundary from particle positions.

### A.5.1 Interpolation method

In the SPEC-debris-barrier platform, the user can select to use the following interpolation methods: bilinear, bicubic, ordinary kriging, and universal kriging. If the kriging interpolation method is selected, the user must specify one of the following semi-variogram models: linear, power, gaussian, spherical, and exponential. The platform uses the Python3 scipy and pykrige libraries to perform the interpolation methods; therefore, the user can refer to online scipy and pykrige libraries for further details. The user selects the interpolation method using:

- *interpolation method*
  - Description: specifies the interpolation method
    - The method recognized by the platform are the following:  
“linear”, “cubic”, “OK svm”, and “UK svm”  
svm = semi-variogram models: “linear”, “power”, “gaussian”, “spherical”, “exponential”
  - Data format: string
  - Example: “linear” or “OK linear” or “UK exponential”
  - Notes:
    - A space is required after the “OK” or “UK” to distinguish the kriging method and the semi-variogram model
    - The “linear” interpolation method is usually selected as the default
  - Analysis usage: *SPEC-debris*, *SPEC-debris-wall*, *SPEC-debris-closed*, *SPEC-debris-combined*, *combined*

### A.5.2 Entrainment model

In the SPEC-debris-barrier platform, two types of entrainment models are used to simulate the effect of debris-flow volume growth, as described in Section 2.2.4.1. The user selects the entrainment model using:

- *entrainment model*
  - Description: specifies one of the two entrainment models:
    - “Hungr” – exponential growth (Eq (2-51))
    - “Er” – compound growth (Eq (2-52))
  - Data format: string
  - Example: “Hungr” or “Er”
  - Notes: the “Hungr” entrainment model is usually selected as the default
  - Analysis usage: *SPEC-debris*, *SPEC-debris-wall*, *SPEC-debris-closed*, *SPEC-debris-combined*, *combined*

### A.5.3 Cluster boundary algorithm

In the SPEC-debris-barrier platform, two algorithms (described in Section 3.3.3.2) can be used to find the debris-flow cluster boundary based on the position of the debris-flow particle: convex-hull and concave-hull algorithm. The user selects the algorithm to compute the cluster boundary through:

- *cluster boundary method*
  - Description: specifies one of the two algorithms (“ConcaveHull” or “ConvexHull”) for computing cluster boundary
  - Data format: string
  - Example: “ConcaveHull” or “ConvexHull”
  - Notes: the “ConcaveHull” algorithm is usually selected as the default
  - Analysis usage: *SPEC-debris*, *SPEC-debris-closed*, *SPEC-debris-combined*

## A.6 Control parameters

### A.6.1 Time

The SPEC-debris analysis uses the prescribed time step interval ( $\Delta t$ ) to incrementally update the particle positions and velocity until the simulation time ( $t$ ) has reached the maximum simulation time ( $t_{max}$ ). The user assigns the time step interval ( $\Delta t$ ) and the maximum simulation time ( $t_{max}$ ) through the following variables:

- *time step interval*
  - Description: time interval between the time steps
  - Data format: float
  - Example: 0.5
  - Analysis usage: *SPEC-debris*, *SPEC-debris-wall*, *SPEC-debris-closed*, *SPEC-debris-combined*
- *maximum simulation time*
  - Description: maximum simulation time
  - Data format: float
  - Example: 10.0
  - Analysis usage: *SPEC-debris*, *SPEC-debris-wall*, *SPEC-debris-closed*, *SPEC-debris-combined*

### A.6.2 Initial particle

The SPEC-debris-barrier platform generates the initial particles (when time step = 0) from the initial soil mass location and depth data from the *source file name*. The initial landslide mass is discretized into  $N$  number of particles. The initial total landslide volume is evenly distributed among the particles arranged in a square spacing pattern. A cell refers to the pixel in DEM data and is bounded between X and Y grids. The initial particles are generated in cells with a non-zero depth (NZD) value, i.e., the initial depth at the cell is larger than zero (0). For example, eight NZD cells are marked by orange boundaries in Figure A.3.

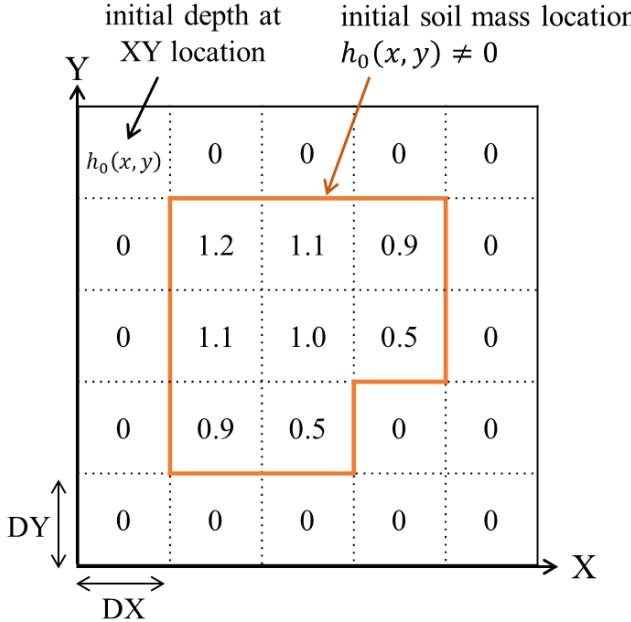


Figure A.3 Source data displaying cells with non-zero depth (NZD) values

One particle is generated per NZD cell by default; therefore, the initial particles' size, volume, and number are determined by DEM grid spacings, i.e., DEM spatial resolutions. A DEM with a lower spatial resolution (larger grid spacings) would produce a fewer number of larger particles. Fewer larger particles may produce less accurate results; therefore, the platform allows the user to generate a higher number of smaller particles by specifying the number of initial particles generated at each cell through the *particle number per cell* variable. However, the user must be aware that a higher number of particles would also increase the computation time or be too demanding for the user's hardware.

- *particle number per cell*
  - Description: number of particles generated per cell
  - Data format: integer
  - Example: 1
  - Notes: a square number in an integer form (e.g., 1, 4, 9, 16, 25) must be specified. The default value is one (1)
  - Analysis usage: *SPEC-debris*, *SPEC-debris-wall*, *SPEC-debris-closed*, *SPEC-debris-combined*

After the number of *particle number per cell* is specified, the platform uses the following procedure to generate the particles positioned at square spacing arrangements:

- (1) locate cells with a non-zero depth (NZD) value (see Figure A.4(a))
- (2) use the k-mean clustering algorithm (Section 3.3.3.1) to classify particles into different source regions (see Figure A.4(b)), and
- (3) subdivide the NZD cells based on the *particle number per cell* variable and place circular particles that fit inside the subdivided NZD cells (see Figure A.4(c)).

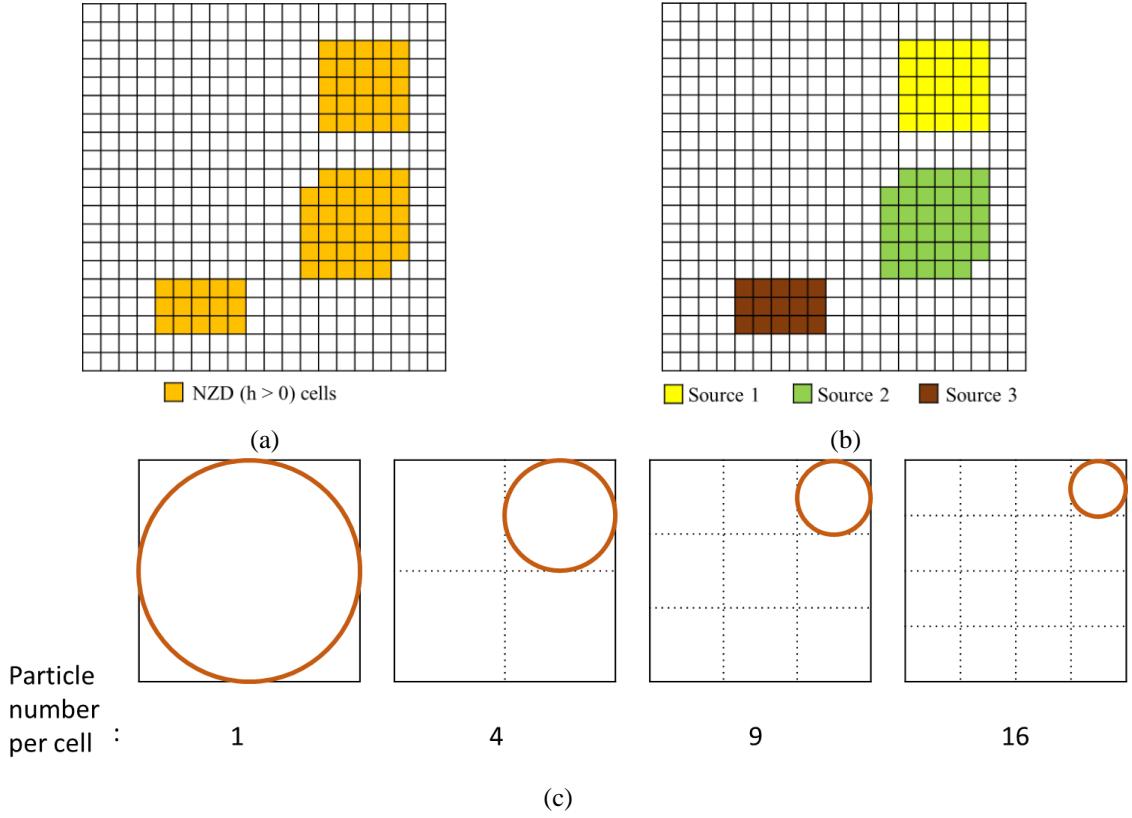


Figure A.4 Initial particle placement: (a) identify NZD cells, (b) classify NZD cells into source regions, (c) subdivide NZD cells and place particles

The initial XY location of the particles is the centroid of the particles. The initial speed ( $u_0$ ) and initial velocity in the X- and Y-directions ( $u_{0x}, u_{0y}$ ) of the particles are determined by the *initial u\_x* and *initial u\_y* variables. If a null data type is assigned to any of the *initial u\_x* and *initial u\_y* variables, the initial velocity is considered zero (0). Otherwise, the initial speed of the particles ( $u_0$ ) is computed as the magnitude of the particle velocity components ( $u_{0x}, u_{0y}$ ) by Eq. (A-4):

$$u_0 = \sqrt{u_{0x}^2 + u_{0y}^2}. \quad (\text{A-4})$$

The initial volume of the particles ( $V_0$ ) are computed by Eq. (A-5):

$$V_0 = \sum(h_{NZD}A_{cell})/N, \quad (\text{A-5})$$

where  $A_{cell}$  is the area of a DEM cell,  $h_{NZD}$  is the initial sliding mass depth at the non-zero depth (NZD) cell specified in the *source file name* data, and  $N$  is the total number of particles. The total number of particles is computed by multiplying the number of NZD cells by the *particle number per cell* value. The initial particle depth ( $h_0$ ) is computed based on the *initial SPH* variable.

### A.6.3 Gravity and free-fall

The gravitational force is the main external driving force influencing the debris-flow propagation.

The acceleration due to gravity is specified with:

- *gravitational acceleration*
  - Description: acceleration caused by gravity
  - Data format: float
  - Example: 9.81
  - Notes: the default value is 9.81 m/s<sup>2</sup>
  - Analysis usage: *SPEC-debris*, *SPEC-debris-wall*, *SPEC-debris-closed*, *SPEC-debris-combined*

In SPEC-debris analysis, the particle is assumed to travel on the surface defined by the DEM data. However, a particle collides against a vertical barrier or suddenly flows over a sudden cliff would be in an air-borne or free-fall state. The gravitational force is the sole force applied to the particle; hence, the energy conservation principle described in Section 2.2.4.6 would ignore the energy loss from rheology. The particle is considered to be in free-fall when the following criteria are met:

$$(90^\circ - \Delta\theta_{free}) \leq |\delta_{dip}| \leq (90^\circ + \Delta\theta_{free}), \quad (\text{A-6})$$

where  $\delta_{dip}$  is the dip angle in the direction of particle movement area in degree unit and  $\Delta\theta_{free}$  is the amount of angle variation at which the particle can be considered to be in a free-fall. The  $\Delta\theta_{free}$  value is specified in the SPEC-debris uses the *free fall angle variation* variable:

- *free fall angle variation*
  - Description: the amount of variation in angle ( $\Delta\theta_{free}$ ) in degree from the vertical (90°) angle at which the particle is considered to be in a free-fall
  - Data format: float
  - Example: 5.0
  - Notes: the default value is 5°
  - Analysis usage: *SPEC-debris*, *SPEC-debris-wall*, *SPEC-debris-closed*, *SPEC-debris-combined*

### A.6.4 Steepest local gradient descent cell size

The pathway algorithm in Section 2.2.3 uses the steepest local gradient descent to determine the slope gradient in the X- and Y-direction in a local DEM region. The local DEM region is defined as an n-by-n local DEM cell region with the centroid cell containing the particle position, as shown in Figure 2.4. The platform specification of two different local DEM cell size  $n$ .

Suppose the terrain surrounding the particle is considered smooth when the elevation slope is gentle with no abrupt elevation changes. Particles in smooth terrain use a larger local cell size  $n$  of five (5) (Hansen et al., 2020) to avoid a particle getting stuck in local minima, such as a shallow ditch, and increase the smoothness of the particle travel path. However, particles nearby a barrier with abrupt elevation changes

should consider the immediate surrounding region; therefore, the smallest local cell size  $n$  of one (1) is used when a particle is detected to be nearby a barrier. The user can modify the default local cell size  $m$  through the *local cell sizes* variable:

- *local cell sizes*
  - Description:
    - defines the local cell size  $n$ . The pathway algorithm in the SPEC-debris model uses the elevation data from an  $n$ -by- $n$  local DEM cell region to determine the steepest local gradient descent.
    - the 1<sup>st</sup> number describes the local cell size  $m$  for particles in smooth terrains
    - the 2<sup>nd</sup> number describes the local cell size  $m$  for particles in non-smooth terrains or nearby a barrier
  - Data format: 1D array with odd integer – [5, 1]
  - Example: [5, 1]
  - Notes: the default value is [5, 1]
  - Analysis usage: *SPEC-debris*, *SPEC-debris-wall*, *SPEC-debris-closed*, *SPEC-debris-combined*

#### A.6.5 SPH smoothing length parameter

The smoothed particle hydrodynamics (SPH) in the SPEC-debris model uses an adaptive smoothing length ( $l$ ) that changes depending on the average density of the neighboring particles. However, a minimum smoothing length factor ( $l_{dp,min}$ ) is specified to limit the smoothing length ( $l$ ) from being too small. According to (Domínguez et al., 2021), the minimum smoothing length factor ( $l_{dp,min}$ ) of two (2) is often used. The user can modify the default minimum smoothing length factor ( $l_{dp,min}$ ) through the *l\_dp\_min* variable:

- *l\_dp\_min*
  - Description: minimum smoothing length factor ( $l_{dp,min}$ ) for computing the smoothing length ( $l$ )
  - Data format: float
  - Example: 2.0
  - Notes: the default value is 2.0
  - Analysis usage: *SPEC-debris*, *SPEC-debris-wall*, *SPEC-debris-closed*, *SPEC-debris-combined*

#### A.6.6 Cluster analysis parameter

The boundary of the debris-flow cluster is defined by the outermost particle locations determined by the cluster boundary algorithm as described in Sections 3.3.3.2. When the concave-hull algorithm is utilized, the  $\alpha$  (alpha) factor determines the shape of the cluster boundary. As the particles propagate, the cluster boundary's shape changes; therefore, the  $\alpha$  factor changes over time. The new  $\alpha$  factor for the cluster is determined iteratively using the bisection method:

- (1) define maximum and minimum  $\alpha$  factors,
- (2) compute the average  $\alpha$  factors of the minimum and maximum  $\alpha$  factors,
- (3) check whether the average  $\alpha$  factors produce a valid boundary shape that contains all the particles inside the boundary,
- (4) replace the maximum  $\alpha$  factor with the average  $\alpha$  factor if the average  $\alpha$  factor does not produce a valid shape; however, replace the minimum  $\alpha$  factor with the average  $\alpha$  factor if the average  $\alpha$  factor produces a valid shape,
- (5) repeat steps (2) to (4) until the following termination criteria are met:
  - a. maximum and minimum  $\alpha$  factors are the same up to 2 decimal places, or
  - b. the number of iterations has exceeded the maximum iteration number

Suppose the bisection method does not converge or terminate due to reaching the maximum iteration number. In that case, the  $\alpha$  factor value is assigned as zero (0), and the convex-hull algorithm is used to determine the cluster boundary. The initial maximum  $\alpha$  factor, the initial minimum  $\alpha$  factor, and the maximum iteration number are specified in the input JSON file with the *concave hull algorithm parameter* variable:

- *concave hull algorithm parameter*
  - Description: specifies the initial maximum  $\alpha$  factor, the initial minimum  $\alpha$  factor, and the maximum iteration number for determining the  $\alpha$  factor in the concave-hull algorithm
  - Data format: object
  - Example:
 

```
“concave hull algorithm parameter”: {  
    “max iteration”: 100,  
    “max alpha”: 0.5,  
    “min alpha”: 0.0  
}
```
  - Notes:
    - The values shown in the example above are the default values
    - “*min alpha*” and “*max alpha*” should be a positive number
  - Analysis usage: *SPEC-debris-closed*, *SPEC-debris-combined*

In the cluster analysis, the merging of two debris-flow clusters is checked by the ratio between the overlapping area, as described in Section 3.3.3.4. The critical overlapping area ratio ( $A_{crit}$ ) is specified through:

- *merge overlap ratio*
  - Description: the critical overlapping area ratio ( $A_{crit}$ ) for determining whether two debris-flow clusters have merged
  - Data format: float
  - Example: 0.5

- Notes:
  - The *merge overlap ratio* value must be between 0 and 1. The default value is 0.5, as described in Cheon et al. (2020)
- Analysis usage: *SPEC-debris-closed*, *SPEC-debris-combined*

### A.6.7 Multiprocessing parameter

The SPEC-debris-barrier platform has incorporated CPU multiprocessing capability that allows full utilization of the CPU processing capacity. The user can specify the number of CPU resources allocated to the SPEC-debris analysis through:

- *max cpu num*
  - Description: maximum number of CPU logical processors used for SPEC-debris analysis
  - Data format: integer
  - Example: 8
  - Notes:
    - the default value is 8; hence, the multiprocessing is used by default
    - if the specified *max cpu num* exceeds the number of CPU logical processors in the computer, then the total number of CPU logical processors is utilized instead of the specified *max cpu num*
  - Analysis usage: *SPEC-debris*, *SPEC-debris-wall*, *SPEC-debris-closed*, *SPEC-debris-combined*

The user should be aware that the larger RAM capabilities are required when using a higher number of CPU logical processors. For example, if the RAM capacity of the computer is small (8 to 16 GB), then it is recommended to assign *max cpu num* to be eight (8) or lower. If the computer has a RAM capability of 64GB or higher, the RAM capacity should no longer be of concern.

The multiprocessing has been tested for AMD and Intel CPU processors; however, the multiprocessing has not been tested for ARM-based CPUs. Therefore, an error may occur if the user wants to run multiprocessing in ARM-based CPUs. If any error occurs while trying to use the CPU multiprocessing, please specify the *max cpu num* to be one (1).

## A.7 Optimal barrier location selection parameters

### A.7.1 Search parameters

The optimal barrier location selection procedure, as described in Section 3.3.5, selects optimal locations for barriers based on the optimization criteria. The efficiency of the barrier location optimization procedure depends on the following factors:

- (1) number of the global loop to avoid the local minima (refer to Section 3.3.5.1),
- (2) determine unsuitable barrier locations (refer to Section 3.3.5.3),
- (3) number of iteration limit to avoid indefinite looping, and

- (4) range of optimal barrier numbers to define a reasonable number of barrier locations.

Based on reasonable engineering judgment, the user can modify these optimization procedure factors to further increase the efficiency of the optimal barrier location selection procedure through the *optimal barrier location selection option* variable:

- *optimal barrier location selection option*
  - Description:
    - Set the factors of the optimal barrier location selection procedure:
      - “*optimal total loop*”: (integer) number of the global loop (refer to Section 3.3.5.1)
      - “*optimal avoid dfl*”: (boolean) option to ignore unsuitable barrier locations, such as location close to the landslide region (refer to Section 3.3.5.3)
      - “*optimal iteration max*”: (integer) maximum number of iterations the search algorithm
      - “*optimal find iteration limit*”: (integer) maximum number of iterations for finding a new set of optimal barrier candidates
      - “*optimal barrier num min*”: (null or integer) minimum number of optimal barrier
      - “*optimal barrier num max*”: (null or integer) maximum number of optimal barrier
  - Data format: object
  - Example:
 

```
“optimal barrier location selection option”: {
    "optimal total loop": 10,
    "optimal avoid dfl": true,
    "optimal iteration max": 500,
    "optimal find iteration limit": 100,
    "optimal barrier num min": null,
    "optimal barrier num max": null
}
```
  - Notes:
    - The values shown in the example is the default value set in the SPEC-debris-barrier platform
    - If the assigned value of “*optimal barrier num min*” is a null type, the minimum number of optimal barriers is one (1)
    - In *SPEC-debris-closed* and *closed* analyses, the default minimum number of optimal barriers is equal to the number of different source regions
    - In *SPEC-debris-combined* and *combined* analyses, the default minimum number of optimal barriers is equal to the closest integer number to 1.5 multiplied by the number of different source regions, i.e., round(1.5 \* “number of landslide source regions”)
  - Analysis usage: *SPEC-debris-closed*, *SPEC-debris-combined*, *closed*, *combined*

## A.7.2 Closed-type barrier optimization

As described in Section 3.3.5.2, the optimal closed-type barrier locations are decided by the set of closed-type barrier locations that give the minimum “cost” (\$). The “cost” is the sum of the product between the weighting factors and the normalized design factors (Eq. (3-19)). The platform requires the user to assign these weighting factors for the optimal closed-type barrier locations selection procedure through:

- *closed barrier optimization criteria*
  - Description:
    - the weighting factors for the design factors considered in the optimal closed-type barrier location selection procedure (refer to Sections 3.3.3.3 and 3.3.5.2)  
“*volume(V)*”: (float) cumulative volume of the debris-flow cluster  
“*pressure(P)*”: (float) average impact pressure of debris-flow cluster  
“*distance\_from\_road(D)*”: (float) distance between the debris-flow cluster centroid to an access route, like a road  
“*closed\_barrier\_number(N)*”: (float) number of closed-type barriers
  - Data format: object
  - Example:

```
“closed barrier optimization criteria”: {  
    "volume(V)": 0.35,  
    "pressure(P)": 0.2,  
    "distance_from_road(D)": 0.15,  
    "closed_barrier_number(N)": 0.3  
}
```
  - Notes:
    - The values shown in the example is the default value set in the SPEC-debris-barrier platform
    - The sum of all weighting factors should be equal to one (1) (Eq. (3-18))
  - Analysis usage: *SPEC-debris-closed, closed*

## A.7.3 Combined open and closed-type barrier optimization

The optimal open-type and closed-type barrier locations, as described in Section 3.3.5, are decided by the set of open and closed-type barrier locations that give the minimum “cost” (\$). The platform requires the user to assign these weighting factors for the optimal combined open and closed-type barrier locations selection procedure through:

- *combined barrier optimization criteria*
  - Description:
    - the weighting factors for the design factors considered in the optimal closed-type barrier location selection procedure (refer to Sections 3.3.3.3 and 3.3.5.2):

- “volume( $V$ )”: (float) cumulative volume of the debris-flow cluster
- “pressure( $P$ )”: (float) average impact pressure of debris-flow cluster
- “distance\_from\_road( $D$ )”: (float) distance between the debris-flow cluster centroid to an access route, like a road

- Data format: object
  - Example:

```

"closed_barrier_optimization_criteria": {
    "volume(V)": 0.35,
    "pressure(P)": 0.2,
    "distance_from_road(D)": 0.15,
    "closed_barrier_number(NC)": 0.2,
    "opened_barrier_number(NO)": 0.1
}

```

- Notes:
    - The values shown in the example is the default value set in the SPEC-debris-barrier platform
    - The sum of all weighting factors should be equal to one (1) (Eq. (3-18))
  - Analysis usage: *SPEC-debris-combined, combined*

One optimal barrier location search procedure is to check whether the open-type barriers have sufficiently lowered the potential risk of debris-flow hazards, as described in Section 3.3.4.4. To analyze the change in the debris-flow risk due to open-type barriers, the user must specify the followings:

- (1) expected performance of the open-type barriers - *open barrier performance*,
  - (2) acceptable level of vulnerability - *critical vulnerability index (VI)*, and
  - (3) vulnerability index model (Kang and Kim, 2014) - *reinforced concrete (RC) wall*

- *open barrier performance*

- #### – Description:

- the expected barrier performances for optimal open-type barriers

“*speed\_ratio(SR)*”: (float) speed reduction ratio (Eq. (3-14))

“*trap\_ratio(TR)*”: (float) volume reduction ratio (Eq. (3-15))

- Data format: object

- Example:

“open barrier performance”: {

“*speed\_ratio(SR)*”: 0.6,

“trap\_ratio(TR)”: 0.6

}

- Notes:
  - The performance ratio value should range between zero (0) and one (1)
- Analysis usage: *SPEC-debris-combined, combined*
- *critical vulnerability index (VI)*
  - Description: accepted level of the debris-flow vulnerability index level
  - Data format: float
  - Example: 0.4
  - Notes: The default value is 0.4, corresponding to the slight damage class (Kang and Kim, 2014)
  - Analysis usage: *SPEC-debris-combined, combined*
- *reinforced concrete (RC) wall*
  - Description: the building structure type affected by the debris-flow
  - Data format: boolean
  - Example: true
  - Notes:
    - The vulnerability function depends on the type of building structure, either reinforced concrete (RC)-framed or non-RC-framed buildings. If true, the affected building is an RC-framed building
  - Analysis usage: *SPEC-debris-combined, combined*

## A.8 Plot options

### A.8.1 Static plot option

The SPEC-debris-barrier platform generates interactive two-dimensional (2D) and three-dimensional (3D) static HTML plots using the Python3 plotly library to display the analysis results. The features and the functions of the interactive static plots are further described in Section 4.4.2. In the input JSON file, the user specifies the options regarding static plots through:

- *static plot option*
  - Description:
    - Set the options for generating interactive 2D and 3D static HTML plots:
      - “*plot map 2D*”: (boolean) option to generate 2D static plot
      - “*plot map 3D*”: (boolean) option to generate 3D static plot
      - “*plot 3D z offset*”: (float) amount of particle position offset in Z-direction in the 3D static plots for the sake of clarity
  - Data format: object
  - Example:
 

```
“static plot option”: { “plot map 2D”: true, “plot map 3D”: true, “plot 3D z offset”: 0.1 }
```
  - Notes:
    - The values shown in the example is the default value set in the SPEC-debris-barrier platform

- The “plot 3D z offset” value only affects the 3D static plot without modifying the computed particle location (XYZ). If the “plot 3D z offset” value is equal to zero (0), the particles appear to be submerged underneath the ground surface; therefore, a slight offset is assigned as the default to show the particles more clearly
- Analysis usage: *SPEC-debris*, *SPEC-debris-wall*, *SPEC-debris-closed*, *SPEC-debris-combined*, *closed*, *combined*

## A.8.2 Animation option

The SPEC-debris-barrier platform also generates interactive 2D and 3D animation HTML plots using the Python3 plotly library to show the SPEC-debris simulation results. The features and the functions of the interactive animation plots are further described in Section 4.4.2. In the input JSON file, the user specifies the options regarding animation plots through:

- *animation option*
  - Description:
    - Set the options for generating interactive 2D and 3D animation HTML plots:
      - “*plot animation 2D*”: (boolean) option to generate 2D animation plot
      - “*plot animation 3D*”: (boolean) option to generate 3D animation plot
      - “*plot animation 2D boundary*”: (boolean) option to plot cluster boundaries at each time-step in the 2D animation plot
      - “*frame duration*”: (integer) duration of an animation frame
      - “*frame transition*”: (integer) duration of an animation transition between the frames
      - “*contour elevation interval*”: (float) option to specify the elevation interval between the contour lines in the 2D animation plot
      - “*animation 3D z offset*”: (float) amount of particle position offset in Z-direction in the 3D animation plots for the sake of clarity
  - Data format: object
  - Example:

```
“animation option”: {
    “plot animation 2D”: true,
    “plot animation 3D”: true,
    “plot animation 2D boundary”: false,
    “frame duration”: 10,
    “frame transition”: 2,
    “contour elevation interval”: 10.0,
    “animation 3D z offset”: 0.1 }
```
  - Notes:
    - The values shown in the example is the default value set in the SPEC-debris-barrier platform

- The unit of “*frame duration*” and “*frame transition*” are milli-seconds (ms)
- The “*animation 3D z offset*” value only affects the 3D animation plot without modifying the computed particle location (XYZ). If the “*animation 3D z offset*” value is equal to zero (0), the particles appear to be submerged underneath the ground surface; therefore, a slight offset is assigned as the default to show the particles more clearly
- Analysis usage: *SPEC-debris*, *SPEC-debris-wall*, *SPEC-debris-closed*, *SPEC-debris-combined*, *closed*, *combined*

### A.8.3 Maximum legend

The interactive plots (both static and animation) have a color-scaled legend displaying the particle properties, such as velocity, depth, and volume. In the input JSON file, the user specifies the maximum value displayed in the color-scaled legends through:

- *max parameter legend*
  - Description:
    - Set the maximum value to be displayed in the color-scale in the interactive HTML plots:
      - “*cluster max velocity(u)*”: (float) maximum cluster velocity legend
      - “*cluster max depth(h)*”: (float) maximum cluster depth legend
      - “*cluster max volume(V)*”: (float) maximum cluster volume legend
      - “*cluster max distance\_from\_road(D)*”: (float) maximum cluster distance-from-road legend
      - “*cluster max pressure(P)*”: (float) maximum cluster impact pressure legend
      - “*particle max velocity(u)*”: (float) maximum particle velocity legend
      - “*particle max depth(h)*”: (float) maximum particle depth legend
      - “*particle max volume(V)*”: (float) maximum particle volume legend
      - “*particle max pressure(P)*”: (float) maximum particle impact pressure legend
  - Data format: object
  - Example: “*max parameter legend*”: {
    - “*cluster max velocity(u)*”: 20,
    - “*cluster max depth(h)*”: 5,
    - “*cluster max volume(V)*”: 10000,
    - “*cluster max distance\_from\_road(D)*”: 300,
    - “*cluster max pressure(P)*”: 500000,
    - “*particle max velocity(u)*”: 20,
    - “*particle max depth(h)*”: 5,
    - “*particle max volume(V)*”: 5,
    - “*particle max pressure(P)*”: 500000 }
  - Analysis usage: *SPEC-debris*, *SPEC-debris-wall*, *SPEC-debris-closed*, *SPEC-debris-combined*, *closed*, *combined*

#### A.8.4 Plot and figure size

The user is required to specify the plot properties through:

- *marker size*
  - Description: the size of point data (e.g., particle) in the interactive plot
  - Data format: integer or float
  - Example: 4
  - Notes:
    - The unit of the *marker size* is pixels
    - The default *marker size* value is 4 pixels
    - As the *marker size* is based on the pixel size, the size would remain the same regardless of the spatial resolution. Hence, the user must be aware that the *marker size* does not represent the actual size of the particles [*marker size* ≠ particle size]
  - Analysis usage: *SPEC-debris*, *SPEC-debris-wall*, *SPEC-debris-closed*, *SPEC-debris-combined*, *closed*, *combined*
- *line width*
  - Description: the thickness of lines (e.g., road) in the interactive plot
  - Data format: integer or float
  - Example: 2
  - Notes:
    - The unit of the *line width* is pixels
    - The default *line width* value is 2 pixels
    - As the *line width* is based on the pixel size, the thickness would remain the same regardless of the spatial resolution
  - Analysis usage: *SPEC-debris*, *SPEC-debris-wall*, *SPEC-debris-closed*, *SPEC-debris-combined*, *closed*, *combined*
- *layout width* and *layout height*
  - Description: the size of the interactive plot in the horizontal and vertical directions
  - Data format: integer or float
  - Example:

```
"layout width": 1000,  
"layout height": 1000,
```
  - Notes:
    - The unit of the *layout width* and *layout height* is pixels
    - The default *layout width* and *layout height* are 1000 pixels and 1000 pixels, respectively
  - Analysis usage: *SPEC-debris*, *SPEC-debris-wall*, *SPEC-debris-closed*, *SPEC-debris-combined*, *closed*, *combined*

## A.9 Output options

### A.9.1 Output file options

The SPEC-debris-barrier platform tabulates the following SPEC-debris simulation results and exports the tabulated results as a csv file to the specified *folder path* location:

- (1) summarized particle data at each time step,
- (2) entire particle class data at each time step,
- (3) summarized cluster data along each flowpath, and
- (4) summary of entire data listing the overall average particle data.

The user can specify whether to generate these tabulated results through:

- *csv output*
  - Description: option to generate tabulated results and export them as csv files
  - Data format: boolean
  - Example: true
  - Notes:
    - the default values is true
    - note that the summarized cluster data along each flowpath is automatically generated
  - Analysis usage: *SPEC-debris*, *SPEC-debris-wall*, *SPEC-debris-closed*, *SPEC-debris-combined*

The platform allows the users to track the progression of optimization search. The barrier sets and their “cost” are displayed at each iteration. Additionally, the ultimate optimal barrier set and their “cost” are displayed on the first line. The progression of the optimization search indicates the efficiency of the optimization search algorithm in the optimal barrier location selection procedure. The progression of the optimization search can be exported as a text (txt) file format through:

- *output optimal step*
  - Description: option to generate the progression of optimization search for optimal barrier location selection procedure and export them as txt files
  - Data format: boolean
  - Example: false
  - Notes: the default values is false
  - Analysis usage: *SPEC-debris-closed*, *SPEC-debris-combined*, *closed*, *combined*

The interactive plots generated are saved on the computer as HTML files that can be opened in any internet browser (e.g., Google Chrome and Microsoft Edge). The following variable allows the platform to open the interactive plots after the files are generated automatically:

- *open plot*
  - Description: option to automatically open the interactive plots on an internet browser

- Data format: boolean
- Example: true
- Notes: the default values is true
- Analysis usage: *SPEC-debris*, *SPEC-debris-wall*, *SPEC-debris-closed*, *SPEC-debris-combined*, *closed*, *combined*

### A.9.2 Decimal points

The user can specify the degree of accuracy stored in output csv files based on the number of decimal points through the *decimal points* variable. It should be noted that the *decimal points* value is applied to all computed values.

- *decimal points*
  - Description: number of decimal points to be stored in output csv files
  - Data format: integer
  - Example: 4
  - Notes: the default value is four (4) decimal places
  - Analysis usage: *SPEC-debris*, *SPEC-debris-wall*, *SPEC-debris-closed*, *SPEC-debris-combined*

### A.10 SPEC-debris cluster

The debris-flow cluster data from the SPEC-debris analysis are stored in the *SPEC\_debris analysis cluster data* variable. The variable specifies a list of the csv output filenames summarizing the cluster data along each flowpath. The results from the cluster analysis are used to generate the flowpath network and perform the optimal barrier location selection procedure described in Section 3.3. In addition, *SPEC\_debris analysis cluster data* variable data are automatically generated when the *SPEC-debris-closed* and *SPEC-debris-combined* analyses are performed. The platform can reuse the simulation results from a SPEC-debris analysis and directly perform closed and combined analysis through this variable.

### A.11 Output JSON

The SPEC-debris-barrier platform generates a JavaScript Object Notation (JSON) file summarizing the analysis inputs and results. The summarized results listed in the output JSON file are the following output variables:

- *particle summary* - summarizes the particle properties from the *SPEC-debris*, *SPEC-debris-wall*, *SPEC-debris-closed*, and *SPEC-debris-combined* analyses
  - “*particle summary*”: {
  - “*total initial volume*”: cumulative particle volume at time-step = 0
  - “*total final volume*”: cumulative particle volume at time-step = max time-step
  - “*max recorded particle velocity*”: maximum velocity of the particles
  - “*max recorded particle depth*”: maximum depth of the particles

- “*max recorded particle pressure*”: maximum impact pressure of the particles
  - }
- *cluster summary* - summarizes the cluster properties from the *SPEC-debris-closed* and *SPEC-debris-combined* analyses
  - “*cluster summary*”: {
    - cluster ID number (integer in string data type): {
      - “*cluster volume at initial*”: cluster volume at time-step = 0
      - “*cluster volume at final*”: cluster volume at time-step = max time-step
      - “*total travel distance*”: cumulative distance travelled by cluster from the source at time-step = max time-step
      - “*max recorded cluster velocity*”: maximum cluster velocity
      - “*max recorded cluster depth*”: maximum depth of the particles
      - “*max recorded cluster pressure*”: maximum impact pressure of the particles
    - },
    - ...
}
- *barrier performance summary* - summarizes the performances of barriers simulated from *SPEC-debris-wall* analysis
  - “*barrier performance summary*”: {
    - barrier ID number (integer in string data type): {
      - “*speed\_ratio(SR)*”: ratio of debris-flow speed before and after barrier collision
      - “*trap\_ratio(TR)*”: ratio of debris-flow volume before and after barrier collision
    - },
    - ...
}
- *optimal barrier location selection summary* - summarizes the optimal barrier location selection analysis results from *SPEC-debris-closed*, *SPEC-debris-combined*, *closed*, and *combined* analyses
  - “*optimal barrier location selection summary*”: {
    - “*optimal closed-type barrier locations*”: lists the DFL id numbers selected as the optimal locations to install closed-type barriers
    - “*optimal open-type barrier locations*”: lists the DFL id numbers selected as the optimal locations to install open-type barriers
    - “*optimal cost*”: the optimization “cost” value of the selected optimal barrier locations
    - “*optimal DFL data*”: summarizes the cluster data at selected DFL locations
    - “*terminus DFL data*”: summarizes the impact of the debris-flow cluster due to colliding with the selected open-type barriers at the terminus locations, i.e., the boundary between the mountain and the urban region}

## Bibliography

- [1] ABAQUS, (2011). “Analysis user’s manual”, *Simulia, Dassault Systems*
- [2] Augarde, C.E. and Lee, S.J. and Loukidis, D. (2021) “Numerical modelling of large deformation problems in geotechnical engineering: A state-of-the-art review.”, *Soils and Foundations*, 61(6), 1718-1735. doi:10.1016/j.sandf.2021.08.007
- [3] Aurenhammer, F. (1991). “Voronoi diagrams—a survey of a fundamental geometric data structure”. *ACM Computing Surveys (CSUR)*, 23(3), 345-405. doi:10.1145/116873.116880.
- [4] Bentley, J. L. (1975). “Multidimensional binary search trees used for associative searching”. *Communications of the ACM*, 18(9), 509-517. doi: 10.1145/361002.361007
- [5] Berti, M., and Simoni, A. (2014). “DFLOWZ: A free program to evaluate the area potentially inundated by a debris flow”. *Computers and Geosciences*, 67, 4– 23. doi:10.1016/j.cageo.2014.02.002.
- [6] Bouscasse, B., Colagrossi, A., Marrone, S., Antuono, M. (2013). “Nonlinear water wave interaction with floating bodies in SPH.” *Journal of Fluids and Structures*, 42, 112-129. doi:10.1016/j.jfluidstructs.2013.05.010.
- [7] Brinkgreve, R.B.J., Swolfs, W.M., Engin, E., Waterman, D., Chesaru, A., Bonnier, P.G., Galavi, V., (2010). PLAXIS 2D 2010 User manual, Plaxis bv.
- [8] Chen HX, Li J, Feng SJ, Gao HY, Zhang DM (2019) “Simulation of interactions between debris flow and check dams on three-dimensional terrain.” Eng Geol 251, 48–62. <https://doi.org/10.1016/j.enggeo.2019.02.001>
- [9] Cheon, E., Lee, S., Lee, S. R., Lee, D., Kim, M., and Lim, H. (2019). “The Framework to Compute Width and Area of Debris-Flow Based on DAN3D”. In *The 32nd KKHTCNN Symposium on Civil Engineering*. KAIST.
- [10] Cheon, E., Lee, S. R., and Lee, D. H. (2020). “Hazard Assessment Based on the Combination of DAN3D and Machine Learning Method for Planning Closed-Type Barriers against Debris-Flow”. *Water*, 12(1), 170.
- [11] Choi, S. K., Kwon, T. H. (2021). “A Case Study on the Closed-Type Barrier Effect on Debris Flows at Mt. Woomyeon, Korea in 2011 via a Numerical Approach.” *Energies*, 14(23), 7890.
- [12] Choi, S. K., Park, J. Y., Lee, D. H., Lee, S. R., Kim, Y. T., Kwon, T. H. (2021). “Assessment of barrier location effect on debris flow based on smoothed particle hydrodynamics (SPH) simulation on 3D terrains”. *Landslides*, 18(1), 217-234.
- [13] Christen, M., Kowalski, J., and Bartelt, P. (2010). “RAMMS: Numerical simulation of dense snow avalanches in three-dimensional terrain”. *Cold Regions Science and Technology*, 63, 1–14. doi:10.1016/j.coldregions.2010.04.005.
- [14] Crozier, M.J. (2010) “Deciphering the effect of climate change on landslide activity: A review.” *Geomorphology*, 124, 260–267, doi:10.1016/j.geomorph.2010.04.009.
- [15] Costa JE. (1984). “Physical geomorphology of debris flows. In Developments and Applications of Geomorphology.” Springer: Berlin Heidelberg; 268–317.
- [16] Cui, P., Zeng, C., Lei, Y. (2015). ”Experimental analysis on the impact force of viscous debris flow”. *Earth Surface Processes and Landforms*, 40(12), 1644-1655.

- [17] Cuomo S, Moretti S, Aversa S (2019) “Effects of artificial barriers on the propagation of debris avalanches”. *Landslides* 16, 1077–1087. <https://doi.org/10.1007/s10346-019-01155-1>
- [18] Dai Z, Huang Y, Cheng H, Xu Q (2017) “SPH model for fluid–structure interaction and its application to debris flow impact estimation”. *Landslides* 14, 917–928. <https://doi.org/10.1007/s10346-016-0777-4>
- [19] De Berg, M. T., Van Kreveld, M., Overmars, M., Schwarzkopf, O. (2000). Computational geometry: algorithms and applications. Springer Science & Business Media.
- [20] Denlinger, R. P., Iverson, R. M. (2004). “Granular avalanches across irregular three-dimensional terrain: 1. Theory and computation”. *Journal of Geophysical Research: Earth Surface*, 109(F1).
- [21] Domínguez, J.M., Fourtakas, G., Altomare, C. et al. (2021) “DualSPHysics: from fluid dynamics to multiphysics problems”. *Comp. Part. Mech.* <https://doi.org/10.1007/s40571-021-00404-2>
- [22] Edelsbrunner, H., Kirkpatrick, D., Seidel, R. (1983). “On the shape of a set of points in the plane”. *IEEE Transactions on information theory*, 29(4), 551-559. doi:10.1109/TIT.1983.1056714.
- [23] FLO-2D Software Inc. 2007. FLO-2D user’s manual. Version 2007.06.
- [24] Gingold, R. A., Monaghan, J. J. (1977). “Smoothed particle hydrodynamics: theory and application to non-spherical stars”. *Monthly notices of the royal astronomical society*, 181(3), 375-389.
- [25] Hansen, K., Hasenstab, K., Schwartzman, A. (2020). “Estimating Mountain Glacier Flowlines by Local Linear Regression Gradient Descent”. *IEEE Transactions on Geoscience and Remote Sensing*, 59(12), 10022-10034.
- [26] Hirt, C. W., Nichols, B. (1988). Flow-3D User's manual (Vol. 107). Flow Science Inc
- [27] Horton, P., Jaboyedoff, M., Rudaz, B. E. A., Zimmermann, M. (2013). “Flow-R, a model for susceptibility mapping of debris flows and other gravitational hazards at a regional scale”. *Natural hazards and earth system sciences*, 13(4), 869-885.
- [28] Hungr, O. (1995). “A model for the runout analysis of rapid flow slides, debris flows, and avalanches”. *Canadian Geotechnical Journal*, 32(4): 610–623. doi:10.1139/t95-063.
- [29] Hungr, O., McDougall, S. (2009). “Two numerical models for landslide dynamic analysis”. *Computers and Geosciences*, 35(5), 978-992.
- [30] HüBl J, Holzinger G. 2003. “Development of design basis for crest open structures for debris flow management in torrents: miniaturized tests for the efficiency estimation of debris flow breakers”. WLS Report 50 Band 3. (2003). Im Auftrag des BMLFUW VC 7a, Vienna: University of Natural Resources and Applied Life Sciences. (in German).
- [31] Itasca Consulting Group, (2019). PFC2D (Particle Flow Code)
- [32] Iverson, R.M., Schilling, S.P., Vallance, J.W. (1998). “Objective delineation of lahar-inundation hazard zones”. *Geological Society of America Bulletin*, 110, 972–984. doi:10.1130/0016-7606(1998)110<0972:ODOLIH>2.3.CO;2;
- [33] Iverson, R. M. (2012). “Elementary theory of bed-sediment entrainment by debris flows and avalanches”. *Journal of Geophysical Research: Earth Surface*, 117(F3).
- [34] Jackson, R. L., Green, I., Marghitu, D. B. (2010). “Predicting the coefficient of restitution of impacting elastic-perfectly plastic spheres”. *Nonlinear Dynamics*, 60(3), 217-229.
- [35] Jakob, M., Hungr, O. (2005). “Debris-flow hazards and related phenomena” Springer, Berlin. Heidelberg, pp. 1-7. doi: 10.1007/3-540-27129-5\_1

- [36] Jeong, S., Lee, K. (2019). “Analysis of the impact force of debris flows on a check dam by using a coupled Eulerian-Lagrangian (CEL) method”. *Computers and Geotechnics*, 116, 103214.
- [37] Kang, H. S., Kim, Y. T. (2014). “Physical vulnerability function of buildings impacted by debris flow”. *Journal of the Korean society of hazard mitigation*, 14(5), 133-143.
- [38] Ketchen, D. J., Shook, C. L. (1996). “The application of cluster analysis in strategic management research: an analysis and critique”. *Strategic management journal*, 17(6), 441-458. doi:10.1002/(SICI)1097-0266(199606)17:6<441::AID-SMJ819>3.0.CO;2-G.
- [39] Kim, M., Lee, S., Kwon, T. H., Choi, S. K., Jeon, J. S. (2021). “Sensitivity analysis of influencing parameters on slit-type barrier performance against debris flow using 3D-based numerical approach”. *International Journal of Sediment Research*, 36(1), 50-62.
- [40] Körner, H.J. (1976). Reichweite und Geschwindigkeit von Bergstürzen und Fließschneelawinen. *Rock Mechanics*, 8: 225–256.
- [41] Korean Design Standard (2018) KDS 51 50 40: 2018
- [42] Kwan JSH, Koo RCH, Ng CWW (2015) “Landslide mobility analysis for design of multiple debris-resisting barriers”. *Can Geotech J* 52, 1345–1359. <https://doi.org/10.1139/cgj-2014-0152>
- [43] Lee, D.H.; Lee, S.R.; Park, J.Y. (2019) “Numerical Simulation of Debris Flow Behavior at Mt. Umyeon using the DAN3D Model.” *J. Korean Soc. Hazard Mitig.*, 19, 195–202, doi:10.9798/KOSHAM.2019.19.3.195.
- [44] Lee, S., An, H., Kim, M., Lim, H. (2020). “Analysis of debris flow simulation parameters with entrainment effect: a case study in the Mt. Umyeon”. *Journal of Korea Water Resources Association*, 53(9), 637-646.
- [45] Li, X., Zhao, J. (2018). “A unified CFD-DEM approach for modeling of debris flow impacts on flexible barriers”. *International Journal for Numerical and Analytical Methods in Geomechanics*, 42(14), 1643-1670.
- [46] Liu J, Nakatani K, Mizuyama T (2013) “Effect assessment of debris flow mitigation works based on numerical simulation by using Kanako 2D”. *Landslides* 10:161–173. <https://doi.org/10.1007/s10346-012-0316-x>
- [47] Lloyd, S. (1982). “Least squares quantization in PCM”. *IEEE transactions on information theory*, 28(2), 129-137. doi:10.1109/TIT.1982.1056489.
- [48] Lucas-Borja, M. E., Piton, G., Nichols, M., Castillo, C., Yang, Y., Zema, D. A. (2019). “The use of check dams for soil restoration at watershed level: A century of history and perspectives”. *Sci. Total Environ*, 692, 37-38.
- [49] Lucy, L. B. (1977). “A numerical approach to the testing of the fission hypothesis”. *The astronomical journal*, 82, 1013-1024.
- [50] Marchi, L., D'Agostino, V. (2004). “Estimation of debris-flow magnitude in the eastern Italian Alps” *Earth. Surf. Process. Landforms*, Vol. 29, pp. 207-220.
- [51] McDougall, S. (2006). “A new continuum dynamic model for the analysis of extremely rapid landslide motion across complex 3D terrain”. Ph.D. thesis, The University of British Columbia, Vancouver, B.C.
- [52] McDougall, S. (2017). “2014 Canadian Geotechnical Colloquium: Landslide runout analysis—current practice and challenges”. *Canadian Geotechnical Journal*, 54(5), 605-620.

- [53] Mergili, M., Fischer, J.-T., Krenn, J., Pudasaini, S.P. (2017). “r.avaflow v1, an advanced open-source computational framework for the propagation and interaction of two-phase mass flows”. *Geoscientific Model Development*, 10: 553–569. doi:10.5194/gmd-10-553-2017.
- [54] Monaghan, J. J. (1985). “Particle methods for hydrodynamics”. *Computer Physics Reports*, 3(2), 71-124.
- [55] Monaghan, J. J. (1989). “On the problem of penetration in particle methods”. *Journal of Computational physics*, 82(1), 1-15.
- [56] Monaghan, J. J. (1992). “Smoothed particle hydrodynamics.” *Annual review of astronomy and astrophysics*, 30(1), 543-574.
- [57] Park, D.W., Lee, S.R., Vasu, N.N., Kang, S.H., Park, J.Y. (2016). “Coupled model for simulation of landslides and debris flows at local scale” *Nat. Hazards*, 81, pp. 1653–1682.
- [58] Park, J.Y., Lee, S.R., Lee, D.H., Choi, G.M. (2018). “A Study of the Combination of Risk Analysis with a Citywide Landslide Early Warning System” *WIT Transactions on Engineering Sciences*, 121, pp. 21-31.
- [59] Pastor, M., Blanc, T., Haddad, B., Petrone, S., Sanchez Morles, M., Drempetic, V., ... others. (2014). “Application of a SPH depth-integrated model to landslide run-out analysis”. *Landslides*, 11(5), 793-812.
- [60] Pedregosa, F., Varoquaux, Ga"el, Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... others. (2011). “Scikit-learn: Machine learning in Python”. *Journal of Machine Learning Research*, 12(Oct), 2825–2830.
- [61] Remaître A, Van Asch TW, Malet JP, Maquaire O (2008) “Influence of check dams on debris-flow run-out intensity”. *Nat Hazards Earth Syst Sci*, 8,1403–1416. <https://doi.org/10.5194/nhess-8-1403-2008>
- [62] Rickenmann, D. (1999). “Empirical relationships for Debris Flow” *Nat. hazards*, Vol. 19, pp. 47-77.
- [63] Rossi, G., Armanini, A. (2019). “Impact force of a surge of water and sediments mixtures against slit check dams”. *Science of the Total Environment*, 683, 351-359.
- [64] Savage, S. B., Hutter, K. (1989). “The motion of a finite mass of granular material down a rough incline”. *Journal of fluid mechanics*, 199, 177-215.
- [65] Scheidl C, Chiari M, Kaitna R, Müllegger M, Krawtschuk A, Zimmermann T, Proske D. (2013). “Analysing Debris-Flow Impact Models, Based on a Small Scale Modelling Approach”. *Survey of Geophysics*, 34(1), 121–140.
- [66] Shen, W., Zhao, T., Zhao, J., Dai, F., Zhou, G. G. (2018). “Quantifying the impact of dry debris flow against a rigid barrier by DEM analyses”. *Engineering Geology*, 241, 86-96.
- [67] Soga, K., Alonso, E., Yerro, A., Kumar, K., Bandara, S. (2016). “Trends in large-deformation analysis of landslide mass movements with particular emphasis on the material point method”. *Géotechnique*, 66(3), 248-273.
- [68] Tiberghien D, Laigle D, Naaim M, Thibert E, Ousset F. (2007). “Experimental investigations of interaction between mudflow and an obstacle. In Debris Flow Hazards Mitigation: Mechanics, Prediction, and Assessment”, Chen CC, Major JJ (eds). Chengdu; 681–687.
- [69] Vassilvitskii, S., Arthur, D. (2006). “k-means++: The advantages of careful seeding”. In *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms* (pp. 1027-1035).

- [70] Vasu, N.N., Lee, S.R. (2016). “A hybrid feature selection algorithm integrating an extreme learning machine for landslide susceptibility modeling of Mt. Woomyeon, South Korea” *Geomorphology*, 263, pp. 50–70.
- [71] Vasu, N.N., Lee, S.R., Pradhan, A.M.S., Kim, Y.T., Kang, S.H., Lee, D.H. (2016). “A new approach to temporal modelling for landslide hazard assessment using an extreme rainfall induced-landslide index” *Eng. Geol.*, Vol. 215, pp. 36–49.
- [72] Voellmy, A. (1955). Über die Zerstörungskraft von Lawinen. *Schweizerische Bauzeitung*, 73, 212–285.
- [73] Vreugdenhil, C. B. (1994). “Numerical methods for shallow-water flow (Vol. 13)”. Springer Science & Business Media.
- [74] H. Wendland (1995). "Piecewise polynomial, positive definite and compactly supported radial functions of minimal degree". *Advances in Computational Mathematics*. 4 (4): 389–396. doi:10.1007/BF02123482. S2CID 36452865.
- [75] Weston.pace; Wikipedia. K Means Example. (2007). Available online: [https://en.wikipedia.org/wiki/K-means\\_clustering#/media/File:K\\_Means\\_Example\\_Step\\_1.svg](https://en.wikipedia.org/wiki/K-means_clustering#/media/File:K_Means_Example_Step_1.svg) (accessed on 1 December 2019).
- [76] Yune, C.Y.; Chae, Y.K.; Paik, J.; Kim, G.; Lee, S.W.; Seo, H.S. (2013) “Debris flow in metropolitan area—2011 Seoul debris flow”. *J. Mt. Sci.*, 10, 199–206, doi:10.1007/s11629-013-2518-7.

## Acknowledgments

First and foremost, I would thank God for guiding my life until now and always being there through everything. Without His help and opening new pathways in my life, I would not have arrived at KAIST and finished my Ph.D. As I have always proclaimed, I hope he will use my degree, skills, and knowledge from now and onwards for God's glory.

I am very grateful to my family. Although some are far from here, they have always supported me spiritually through prayers and encouraged me to pursue my goals. I would especially thank my grandmother and relatives for living in South Korea. They have always welcomed me into their homes and helped me all these times.

I am very thankful to have the opportunity to be a student of Prof. Seung-Rae Lee, a fantastic advisor and an excellent mentor. His gentle nature and charismatic leadership have hugely inspired me. Additionally, his technical yet practical approach to research has greatly inspired my approach to Ph.D. research. Thank you to my Ph.D. committee members, Prof. Gye-Chun Cho, Prof. Tae-Hyuk Kwon, Prof. Jinhyun Choo, and Prof. Yun-Tae Kim, for all the guidance and discussions that led to many improvements. I hope that I may learn their research insights for future research endeavors.

I am grateful to all my seniors, colleagues, and juniors at KAIST. I thank my research mentors, Dr. Deuk-Hwan Lee and Dr. Jun-Young Park, for their valuable insight and inspiration in research; my seniors, Dr. Shingyu Choi and Dr. Min-Seop Kim, for all the help and encouragement; Dr. Hwan-Hui Lim for all the help and being a big brother for me; Captain Seung-min Lee for encouraging me to stay healthy and sharp; Ye-Eun Nam for being a cheerful and bright presence; Ryan Ramirez, Satish Manandhar, Yidong Zhao, Ramos Oliver Jr., Dong-Yeup Park, June-Ho Park, and Woo-Jae Jang, for always being great friends and colleagues; and, last but not least, Byeol Nim Cha, Mun Suk Jeong, and Chan Seon Lee for helping with the administration works. Additionally, I would like to thank our research colleagues in PKNU, Ji-Sung Lee, Hyeong-Jin Kim, and Chang-Ho Song, for many accompaniments during the research. I hope everyone stays physically, mentally, and spiritually healthy as they aim to achieve their visions.

I would like to thank my friends, especially Elisha Lai and Daniel Kim, from my Hebron School; friends I made in UK and USA, especially Matthew Law; my friends and family in faith at Daejeon and Seoul, especially Dr. Minhye Zoh, Dr. Minjae Zoh, David Zoh, Cheung Ji Yoon, and Seung Hoon Im, for all the prayers and your friendships.

June 2022

Yours sincerely,

Enok Cheon

# Curriculum Vitae

Name: Enok Cheon

Date of Birth: April 13<sup>th</sup>, 1995

## Educations

2019. 8. - 2022. 8. Ph.D. in Civil and Environmental Engineering, KAIST, Republic of Korea

2015. 8. - 2016. 5. Exchange Student, University of Illinois at Urbana-Champaign (UIUC), USA

2013. 9. - 2017. 6. M.Eng. in Civil and Environmental Engineering, University College London (UCL), UK

## Career and Services

2019. 8. – 2022. 8. Research/teaching assistant at the Department of Civil and Environmental Engineering, KAIST

2019. 3. – 2019. 7. Contract researcher at the Department of Civil and Environmental Engineering, KAIST

2017. 8. – 2018. 8. Research assistant at the University of Illinois at Urbana-Champaign, UIUC

2016. 5. - 2016. 9. Research internship at the University of Illinois at Urbana-Champaign, UIUC

2009. 8. - 2011. 6. Tutor and mentor for TRIO Upward Bound College Prep Academy at the University of Illinois  
at Urbana-Champaign, UIUC

2014. 9. – 2014. 12. Student mentor in Civil and Environmental Engineering, University College London (UCL)

2013. 8. – 2014. 9. Architecture Global Brigade volunteering in Honduras

## Professional Certificate and Awards

2021. Silver award, Geoscience and Mineral Resources, and Geo Artificial Intelligence Competition (KIGAM)

2017-2018. Olson Fellowship, University of Illinois Urbana-Champaign (UIUC)

2013. UCL Provost's Excellence Scholarship, University College London (UCL)

2013. The Duke of Edinburgh Award (International Award for Young People) - Gold level

2013. Scholar of the year, Hebron School

2012. DipABRSM for piano performance, ABRSM board

2010. ABRSM Grade 8 practical examination for violin performance, ABRSM board

## Research Achievements

### International Journals (SCI/SCIE)

- [1] Min-Seop Kim, Seung-Rae Lee, **Enok Cheon**, Min-Jun Kim, Seok Yoon. (2021). “Thermochemical changes on swelling pressure of compacted bentonite.” *Annals of Nuclear Energy*, 151, 107882.
- [2] Deuk-Hwan Lee, **Enok Cheon**, Hwan-Hui Lim, Yun-Tae Kim, Seung-Rae Lee (2020). “Artificial neural network as a prediction model for debris-flow volume caused by extreme rainfall in the central region of South Korea”, *Engineering Geology*, under review.
- [3] Hwan-Hui Lim, **Enok Cheon**, Deuk-Hwan Lee, Jun-Seo Jeon, Seung-Rae Lee (2020). “Classification of Granite Soils and Prediction of Soil Water Content Using Hyperspectral Visible and Near-Infrared Imaging”, *Sensors*, Vol. 20, pp. 1-15.
- [4] **Enok Cheon**, Seung-Rae Lee, Deuk-Hwan Lee (2020). “Hazard Assessment Based on the Combination of DAN3D and Machine Learning Method for Planning Closed-Type Barriers against Debris-Flow”, *Water*, Vol. 12, pp. 1-24.

### Domestic Journals (KCI)

- [1] 임환희, 이승래, 전예녹, 이득환 (2019). “초분광 근적외선 영상기술을 이용한 흙의 험수비 측정 기술”, *한국지반공학회 논문집*, Vol. 35(11), pp. 51-62.

### International Conference Proceedings

- [1] **Enok Cheon**, Seung-Rae Lee, Jinyun Choo, Hwan-Hui Lim, and Ye-Eun Nam (2022). “Development of Raspberry Pi-based IoT Landslide Monitoring System”, *2<sup>nd</sup> EAGE/SEG Workshop on Geophysical Aspects of Smart Cities 2022*, Hong Kong (Under review).
- [2] **Enok Cheon**, Seung-Rae Lee, Deuk-Hwan Lee, Hwan-Hui Lim, and Seung-Min Lee (2020). “Combination of Web Programming and DAN3D to Generate Interactive Plots for Debris Flow Hazard Assessment”, *EGU General Assembly 2020*, Vienna, Austria (Geoscience Online).
- [3] Hwan-Hui Lim, **Enok Cheon**, Deuk-Hwan Lee, Seung-Min Lee, and Seung-Rae Lee (2020). “Soil Water Content Variation Regression Analysis Using Hyperspectral Camera in Weathered Granite Soils”, *EGU General Assembly 2020*, Vienna, Austria (Geoscience Online).
- [4] **Enok Cheon**, Seung-Min Lee, Seung-Rae Lee, Deuk-Hwan Lee, Min-Seop Kim, and Hwan-Hui Lim (2019). “A Framework to Compute the Width and Area of Debris-Flow Based on DAN3D”, *The Thirty-Second KKHTCNN Symposium on Civil Engineering*, Daejeon, Korea, 24-26 October 2019.
- [5] Min-Seop Kim, Seung-Rae Lee, **Enok Cheon**, Min-Jun Kim, Seok Yoon. (2019). “Influence of temperature on swelling pressure in Korean domestic bentonite.” *The Thirty-Second KKHTCNN Symposium on Civil Engineering*, Daejeon, Korea, 24-26 October 2019.
- [6] **Enok Cheon**, Seung-Rae Lee, Deuk-Hwan Lee, Hwan-Hui Lim and Seung-Min Lee (2019). “A Framework for Designing Slit-Type Barrier for Debris-Flow Hazards”, *12<sup>th</sup> ARC of IAEG*, Jeju Island, Korea.

#### Domestic Conference Proceedings

- [1] 임환희, 이승래, 전예녹, 이득환, 이승민 (2020). “초분광 카메라를 이용한 화강풍화토의 힘수비 변화관련 매개변수” 2020 대한토목학회 컨벤션, 대한민국 제주.
- [2] 김민섭, 이승래, 전예녹, 윤석. (2020). “Influence of Temperature on Mechanical Properties of Korean Compacted Bentonite.” In 대한토목학회 2020 컨벤션. 대한토목학회.

#### Domestic Patents Application

- [1] 다중 영향인자를 고려한 토석류 위험유역 내 폐쇄형 사방댐의 최적 위치선정 방법, NO. 10-2020-0131196

#### Copyright Registration

- [1] 토석류 증강현실 Debris-Flow AR, 프로그램 등록, 2019-11-27, No. C-2019-036088