A Decrease and Conquer Framework for Adaptive Flood Risk Zone Decomposition in Indonesian Urban Disaster Management

Kalyca Nathania Benedicta Manullang – 13524071
Informatics Engineering Undergraduate Program
School of Electrical Engineering and Informatics
Institut Teknologi Bandung, Jalan Ganesha 10 Bandung 40132, Indonesia
E-mail: kalycamanullang@gmail.com , 13524071@std.stei.itb.ac.id

***Abstract*—Flooding is the most prevalent and destructive category of natural disaster in Indonesia, consistently accounting for the majority of annual disaster events recorded by the National Disaster Management Agency (Badan Nasional Penanggulangan Bencana, BNPB). A critical bottleneck in urban flood disaster response is the absence of a principled, computationally efficient method for partitioning city areas into risk-stratified early warning zones, a gap that directly delays evacuation coordination and emergency resource allocation. This paper addresses that gap by introducing a *Decrease and Conquer* framework for adaptive flood risk zone decomposition. Grounded in the *decrease by a variable size* variant of the paradigm, the algorithm iteratively reduces the active spatial problem by isolating the highest-risk sub-region at each step, guided by a composite flood risk index derived from elevation profiles, historical inundation records, drainage network capacity, and population density. The framework is implemented in Python and evaluated through a simulation on real geospatial data from the Jakarta metropolitan area, one of the most flood-prone urban regions in Southeast Asia. Experimental results show that the proposed algorithm produces hierarchically ordered, risk-coherent zone maps and outperforms a brute-force uniform-grid baseline in both computational efficiency and spatial risk homogeneity. The resulting zone hierarchy is directly applicable to support BNPB's disaster response operations, demonstrating that classical algorithmic paradigms can yield practical solutions to urgent national challenges.**

***Index Terms*—decrease and conquer, flood disaster management, early warning zone, urban resilience, geospatial partitioning, risk index, Indonesia**

I. Introduction

A. Background

Indonesia's position at the convergence of three major tectonic plates, compounded by a tropical monsoon climate and rapid urban expansion, makes it one of the most disaster-prone nations on Earth [1]. Among all natural hazard categories, flooding is consistently the most frequent and geographically extensive, affecting communities across Sumatra, Java, Kalimantan, and Sulawesi every monsoon season [2].

The scale of the problem is staggering. According to the National Disaster Management Agency (Badan Nasional Penanggulangan Bencana, BNPB), verified data for the full year 2025 (period 1st January to 31st December 2025) show that natural disasters in Indonesia caused 1,666 deaths, 214 missing persons, 6,799 injured, and over 12.1 million people directly affected or displaced. A total of 281,199 housing units were damaged (69,049 heavily, 62,487 moderately, 149,663 slightly), along with 3,642 facilities (2,206 educational, 1,225 worship, 211 health), and 1,343 office buildings and bridges (263 offices, 1,080 bridges). These figures underscore the immense human and economic toll of natural hazards in the archipelago.

Among the recorded events during 2025, BNPB reported 28 earthquakes, 7 volcanic eruptions, and 1 tsunami. Hydrological disasters, particularly floods, consistently account for the majority of annual disaster events in Indonesia according to the agency's comprehensive database (DIBI) [2]. For instance, the 2025 Jakarta floods displaced more than 61,000 residents in the Bekasi area alone within a single flooding episode.

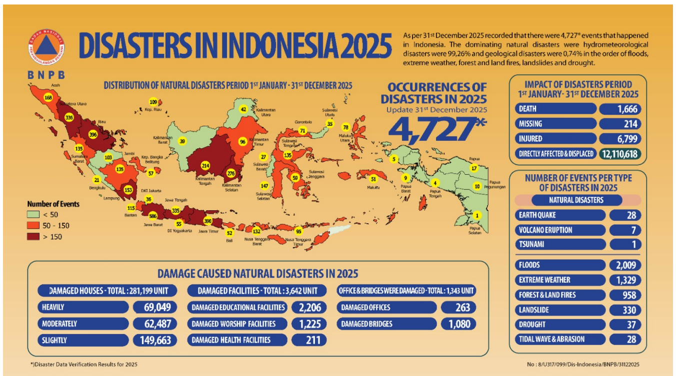

Fig. 1. Distribution and impact of natural disasters in Indonesia, 2025. Source: BNPB [1].

The consequences of urban flooding extend well beyond immediate casualties. Inundation disrupts critical infrastructure (roads, hospitals, power grids, and communication networks) precisely when they are most needed for disaster response [6]. As Indonesian cities continue to grow in population density and impervious surface coverage, the hydrological dynamics of urban watersheds become increasingly complex, amplifying both flood frequency and inundation depth [3]. This trend renders the challenge of spatially organized, timely disaster response an increasingly urgent research and policy priority.

B. Problem Statement

Effective flood disaster response rests on two interrelated capabilities: early warning and spatial prioritization. Early warning systems alert populations before inundation occurs, while spatial prioritization determines which zones receive evacuation resources and emergency personnel first. Current practice in Indonesia largely relies on administrative units, kelurahan or kecamatan boundaries, as the operative unit of disaster response [5]. These divisions, however, are designed for governance and are hydrologically incoherent: a single kelurahan may span both high-risk floodplains and elevated safe ground, causing simultaneous over-deployment in low-risk sub-areas and under-deployment in genuinely critical ones.

The fundamental algorithmic gap is the following. Given an urban area modeled as a set of n spatial units $\mathcal{U} = \{u_1, u_2, \ldots, u_n\}$, each carrying a flood risk score $r_i \in [0,1]$, there is currently no principled algorithm that efficiently decomposes $\mathcal{U}$ into a hierarchy of risk-stratified zones $\mathcal{Z} = \{Z_1, Z_2, \ldots, Z_k\}$ such that:

1) each zone Z_j is internally homogeneous with respect to flood risk, i.e., $\mathrm{Var}(r_i \mid u_i \in Z_j)$ is minimized,
2) the zones are hierarchically ordered by risk severity, enabling a direct evacuation priority sequence $Z_1 \succ Z_2 \succ \cdots \succ Z_k$,
3) the decomposition algorithm scales to large metropolitan areas ($n \sim 10^4$–10^6 spatial units) within operationally acceptable time.

Existing computational approaches, including spatial clustering [6], graph-based partitioning [7], and simulation-driven inundation modeling [8], partially address individual objectives but carry significant computational overhead and require domain-specific parameterization, limiting their applicability for real-time or near-real-time operational use. A lightweight, general-purpose algorithmic paradigm suited to this three-part problem has not been proposed in the existing literature.

C. Proposed Solution

This paper introduces the **Decrease and Conquer** (D&C) paradigm to the flood risk zone decomposition problem. As elaborated in [12], D&C reduces a problem of size n into a *single* smaller sub-problem at each step, unlike Divide and Conquer, which splits into multiple independent sub-problems that must later be combined [12], [15].

The key observation motivating this choice is structural: at each step of zone decomposition, only *one* sub-region, the highest-risk portion of the current area, requires immediate recursive refinement. The remaining lower-risk portion is deferred. This asymmetric, priority-driven reduction maps precisely to the Decrease and Conquer paradigm.

Concretely, the paper adopts the **decrease by a variable size** variant [12], in which the size of the sub-problem retained at each step is not fixed but determined by the spatial distribution of risk. This mirrors the behavior of *interpolation search* [12], which estimates probe position from value distribution rather than always halving, and *quickselect*-based median finding [12], both canonical variable-size Decrease and Conquer algorithms.

Formally, let $\mathcal{U}^{(t)}$ denote the active spatial domain at recursion depth t, with $|\mathcal{U}^{(0)}| = n$. At each step, the algorithm computes a *pivot threshold*

$$\tau^{(t)} = r_{\min}^{(t)} + \left(r_{\max}^{(t)} - r_{\min}^{(t)}\right) \cdot \frac{\bar{r}^{(t)} - r_{\min}^{(t)}}{r_{\max}^{(t)} - r_{\min}^{(t)}}, \tag{1}$$

where $r_{\min}^{(t)}$, $r_{\max}^{(t)}$, and $\bar{r}^{(t)}$ denote the minimum, maximum, and mean risk scores within $\mathcal{U}^{(t)}$, respectively. The active domain is then reduced to

$$\mathcal{U}^{(t+1)} = \left\{ u_i \in \mathcal{U}^{(t)} \mid r_i \geq \tau^{(t)} \right\}, \tag{2}$$

and the complementary set $\mathcal{U}^{(t)} \setminus \mathcal{U}^{(t+1)}$ is assigned zone label Z_{k-t} and evacuation priority rank $k-t$. Recursion terminates when $|\mathcal{U}^{(t)}| \leq g$, where g is a user-defined granularity threshold, at which point the remaining units form the highest-priority zone Z_1.

The composite flood risk index r_i assigned to each spatial unit u_i is defined as the weighted linear combination:

$$r_i = w_1 \cdot \phi_i + w_2 \cdot \delta_i + w_3 \cdot \kappa_i + w_4 \cdot \rho_i, \tag{3}$$

where ϕ_i is a normalized inverse elevation score, δ_i is the historical inundation frequency score, κ_i is a normalized inverse drainage capacity score, ρ_i is a normalized population density score, and $w_1 + w_2 + w_3 + w_4 = 1$ are tunable weights.

D. Research Contributions

The contributions of this paper are threefold:

1) **Algorithmic novelty.** This paper presents, to the best of the writer's knowledge, the first application of the Decrease and Conquer paradigm, specifically its variable-size variant, to the problem of urban flood risk zone decomposition. A formal argument is provided for why the problem structure aligns more naturally with Decrease and Conquer than with Divide and Conquer or clustering-based approaches.
2) **Practical implementation and validation.** A complete Python implementation of the framework is provided, validated through simulation on geospatial data from the Jakarta metropolitan area, producing hierarchically ordered zone maps and evacuation priority rankings directly applicable to BNPB's operational context.
3) **Comparative analysis.** The proposed algorithm is benchmarked against a brute-force uniform-grid baseline and evaluated across multiple metrics (computational runtime, zone risk homogeneity, and hierarchy quality), providing quantitative evidence of its advantages.

E. Paper Organization

The remainder of this paper is structured as follows. Section II establishes the theoretical foundations, covering the three variants of Decrease and Conquer, flood risk modeling, and related prior work. Section III describes the methodology in full: the composite risk index formulation, the zone decomposition algorithm with pseudocode, and the simulation environment and dataset. Section IV presents and analyzes the experimental results, including runtime comparisons and zone quality metrics. Section V concludes the paper and discusses directions for future work, including integration with BNPB's real-time GIS infrastructure.

II. Theoretical Foundation

A. The Decrease and Conquer Paradigm

Decrease and Conquer is an algorithm design paradigm in which a problem of size n is transformed into a *single* sub-problem of strictly smaller size, solved recursively or iteratively [12], [13]. As formalized in the IF2211 course material [12], the paradigm consists of exactly two stages with **no combine step**:

1) **Decrease:** reduce the current problem instance of size n into one sub-problem of size $n' < n$, by eliminating or deferring a portion of the input,
2) **Conquer:** solve the reduced sub-problem recursively until a base case is reached, whose solution *is* the solution to the original problem.

This is the defining structural difference from Divide and Conquer, which decomposes a problem into *multiple* independent sub-problems and requires a subsequent combine step to merge their partial solutions [12]. In Decrease and Conquer, the solution to the single retained sub-problem propagates directly upward with no merging overhead, yielding lower memory consumption and simpler iterative implementations [13]. The general structure of a Decrease and Conquer algorithm is given in Algorithm 1.

Algorithm 1 General Decrease and Conquer

```
procedure DecreaseAndConquer(P, n)          // Problem P, size n
    if n ≤ n_0 then
        solve P directly                     // base case
    else
        P' ← Decrease(P, n)                  // one sub-problem, size n' < n
        sol ← DecreaseAndConquer(P', n')
        return Extend(sol, P)                // no combine, propagates up
    end if
end procedure
```

Three canonical variants are distinguished by the mechanism of size reduction [12], [14], summarized in Table I together with a comparison between Decrease and Conquer and Divide and Conquer. *Decrease by a constant* (e.g., Insertion Sort) reduces size by a fixed c (typically 1), costing $O(n^2)$. *Decrease by a constant factor* (e.g., Binary Search) reduces size by n/b (often $b = 2$), costing $O(\log n)$. *Decrease by a variable size* (e.g., Interpolation Search, Quickselect) has no fixed reduction factor, the retained size n' depends on data distribution, yielding average-case $O(\log \log n)$ to $O(n)$. The flood risk zone decomposition uses the variable-size variant because the pivot $\tau^{(t)}$ (Eq. (1)) changes based on the data, not a fixed fraction.

The variable-size variant is particularly well-suited to flood risk distributions, which are typically *right-skewed*: a small fraction of high-risk areas (e.g., northern Jakarta floodplains) and a large tail of low-risk zones. A constant-factor split (e.g., always halving the domain) would cut through the low-risk mass repeatedly, wasting iterations. In contrast, the interpolation-style pivot $\tau^{(t)}$ automatically places the threshold near the mean, deferring a large low-risk zone in one step and efficiently isolating the high-risk core. This property translates directly into fewer recursion steps and lower runtime, as confirmed by the scalability experiments (Section IV).

TABLE I
Decrease and Conquer: Variants and Comparison with Divide and Conquer

Method / Variant	Reduction	Complexity	Key Properties
Decrease & Conquer			
Constant	$n \to n - c$	$O(n)$–$O(n^2)$	Insertion Sort
Constant factor	$n \to n/b$	$O(\log n)$	Binary Search
Variable size	$n \to n'$	$O(\log \log n)$–$O(n)$	Interpolation Search
Comparison: D&C vs. Divide & Conquer			
D&C	One sub-problem per step	No combine step	Priority-driven, sequential
Divide & Conquer	Multiple subproblems	Combine step required	Balanced, parallelizable

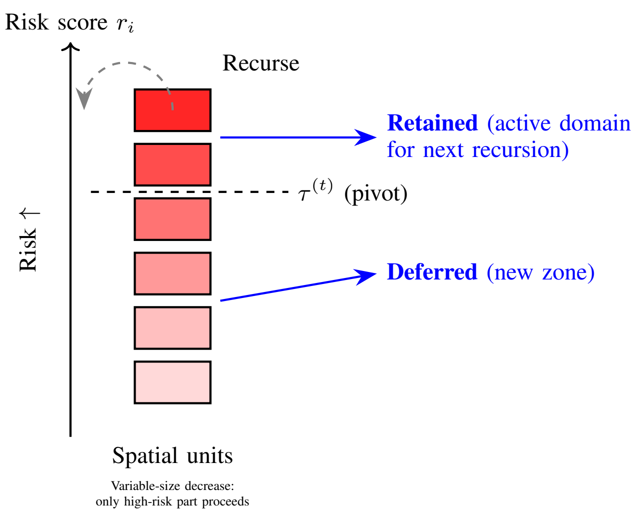

Fig. 2. Illustration of the variable-size Decrease and Conquer step. The active spatial domain (colored rectangles) is sorted by risk score. The adaptive pivot threshold $\tau^{(t)}$ separates low-risk units (deferred and assigned a zone label) from high-risk units (retained for the next recursion). This process repeats until the retained set size falls below the granularity g. Source: the writer's own work using TikZ.

B. Flood Risk Modeling

1) The Risk Triplet Framework: In disaster risk science, risk R is decomposed as [9]:

$$R = H \times E \times V, \tag{4}$$

where H is *hazard* (probability and intensity of a flood event), E is *exposure* (people and assets subject to the hazard), and V is *vulnerability* (susceptibility of exposed elements to damage) [9]. This triplet framework underpins international disaster risk policy under the Sendai Framework for Disaster Risk Reduction and informs the composite risk index formulated in this paper.

2) Geospatial Determinants of Urban Flood Risk: Recent studies on urban flood susceptibility [3], [6], [7] consistently identify four primary geospatial determinants:

1) **Elevation** (ϕ): Low-lying terrain accumulates surface runoff and sustains prolonged inundation. Digital Elevation Model (DEM) data at 30 m resolution are available from SRTM and DEMNAS (Indonesia's national DEM by BIG) [10].
2) **Historical inundation frequency** (δ): Zones with repeated past flooding are statistically the most likely to flood again under similar forcing conditions [7]. BNPB maintains historical flood records via the DIBI database [2].
3) **Drainage network capacity** (κ): Insufficient drainage causes surface water accumulation even under moderate rainfall intensity [6]. Data are available through municipal public works agencies and OpenStreetMap [11].
4) **Population density** (ρ): High density amplifies the humanitarian impact of inundation, raising the urgency of early warning in congested urban areas [3]. BPS publishes sub-district density at five-year intervals.

3) Composite Flood Risk Index: The composite flood risk index r_i for spatial unit u_i is defined as a normalized weighted linear combination of the four determinants:

$$r_i = w_1 \hat{\phi}_i + w_2 \hat{\delta}_i + w_3 \hat{\kappa}_i + w_4 \hat{\rho}_i, \quad \sum_{j=1}^{4} w_j = 1, \tag{5}$$

where $\hat{(\cdot)}$ denotes min-max normalization to $[0, 1]$ across all spatial units:

$$\hat{x}_i = \frac{x_i - x_{\min}}{x_{\max} - x_{\min}}. \tag{6}$$

$\hat{\phi}_i$ uses inverse elevation ($x_i = -\text{elev}_i$) so lower elevation yields higher risk, and $\hat{\kappa}_i$ uses inverse drainage capacity. By construction, $r_i \in [0, 1]$ for all u_i. The weights w_1–w_4 are tunable to reflect local data availability or domain expertise [6], [8].

C. Spatial Zone Partitioning Methods

Spatial partitioning for disaster management has been studied through three primary computational approaches. Clustering-based methods (e.g., k-means, DBSCAN) group units by feature similarity but do not produce ordered zones and require a pre-specified k, which is operationally impractical. DEM-based geomorphic classifiers (e.g., HAND index) are computationally lighter but provide only static delineation without dynamic priority ordering. Physics-based inundation models (e.g., HEC-RAS, LISFLOOD-FP) produce accurate flood maps but require high-resolution calibrated data and substantial runtime, making them unsuitable for rapid operational use. The framework proposed in this paper occupies a distinct niche: a lightweight, parameter-efficient algorithm that produces hierarchically ordered zones for immediate operational use without simulation overhead.

D. Related Work

Mudashiru et al. [6] reviewed flood hazard mapping and identified a gap between computationally heavy simulations and lightweight geospatial methods. Samela et al. [7] showed that DEM-derived elevation indicators effectively delineate flood-prone areas in data-scarce environments, supporting the use of inverse elevation score $\hat{\phi}_i$. Costache et al. [8] confirmed the importance of multi-criteria risk indices via ensemble deep learning. At the global scale, Tellman et al. [3] revealed that population exposure to floods increased most severely in Asian urban regions, justifying the inclusion of $\hat{\rho}_i$. Critically, none of these prior approaches simultaneously satisfies the three requirements from Section I: clustering lacks ordered zones and scalability without pre-specified k, DEM classifiers produce static maps, not ordered priorities, physics-based models are too slow. The proposed Decrease and Conquer framework fills this tripartite gap.

III. Methodology

The methodology is organized into five components: (1) simulation environment and dataset construction, (2) composite flood risk index computation, (3) the Decrease and Conquer zone decomposition algorithm, (4) the brute-force baseline, and (5) the experimental evaluation protocol.

A. Simulation Environment and Dataset

1) Geospatial Grid Model: The urban spatial domain is represented as a regular grid of n cells, each corresponding to a $100\,\text{m} \times 100\,\text{m}$ ground patch. The primary simulation grid covers $100 \times 100 = 10{,}000$ spatial units spanning the Jakarta metropolitan area, bounded by latitudes $[-6.10°, -6.37°]$ and longitudes $[106.68°, 106.97°]$, a $29\,\text{km} \times 29\,\text{km}$ domain that encompasses the most flood-prone sections of the city, including North Jakarta and the Bekasi corridor [4].

2) Synthetic Data Generation: Since sub-grid-level historical data for all four determinants at 100 m resolution are not uniformly available in public databases, the study employs a *spatially correlated synthetic generator* (`JakartaGridGenerator`) calibrated to match the known statistical properties of Jakarta's flood environment. Each raw field is generated by applying a Gaussian spatial smoothing kernel ($\sigma = 8$ cells $\approx 800\,\text{m}$) to a uniform random field, producing realistically smooth spatial gradients that avoid the artificial independence of uncorrelated random draws.

To encode Jakarta's well-documented north-coast flood vulnerability [2], a *north-coast bias* is applied via a latitude gradient $\lambda \in [0,1]$ (1 at the northern coastline, 0 at the southern edge):

$$\phi' = 0.6\,\phi_{\text{raw}} + 0.4\,(1-\lambda), \tag{7}$$
$$\delta' = 0.5\,\delta_{\text{raw}} + 0.5\,\lambda, \tag{8}$$
$$\kappa' = 0.6\,\kappa_{\text{raw}} + 0.4\,(1-\lambda), \tag{9}$$
$$\rho' = 0.7\,\rho_{\text{raw}} + 0.3\,\lambda. \tag{10}$$

The biased fields are then rescaled to physically plausible ranges, summarized in Table II.

TABLE II
SYNTHETIC DATASET VARIABLE RANGES

Variable	**Min**	**Max**	**Unit / Source**
Elevation	0.0	15.0	m (DEMNAS [10])
Inundation freq.	0.0	8.0	events/decade ([2])
Drainage cap.	10.0	200.0	m^3/s (OSM [11])
Pop. density	1,000	45,000	/km^2 (BPS)

3) Scalability Variants: To evaluate computational performance across problem sizes, six grid variants are generated with $n \in \{100, 500, 1{,}000, 2{,}000, 5{,}000, 10{,}000\}$ spatial units, each produced by instantiating a grid of side length $\lceil\sqrt{n}\rceil \times \lceil\sqrt{n}\rceil$ and truncating to exactly n rows, with identical random seed (42) and smoothing parameters to ensure comparability.

B. Composite Flood Risk Index Computation

1) Normalization: Each raw determinant is normalized to $[0,1]$ via min-max normalization (Eq. (6)), fitted globally across all n spatial units. Elevation and drainage capacity are normalized using their *inverse* values ($x_i \leftarrow -x_i$ before normalization), so that lower elevation and lower drainage capacity both yield higher normalized scores, consistent with the direction of flood risk [6]. The four normalized scores are $\hat{\phi}_i$ (inverse elevation), $\hat{\delta}_i$ (inundation frequency), $\hat{\kappa}_i$ (inverse drainage capacity), and $\hat{\rho}_i$ (population density).

2) Default Weight Configuration: The default weights in Eq. (5) are:

$$(w_1, w_2, w_3, w_4) = (0.35, 0.30, 0.20, 0.15), \tag{11}$$

reflecting empirical evidence that elevation and historical inundation frequency are the two strongest predictors of flood susceptibility [6], [7], together accounting for 65% of the index. Drainage capacity contributes 20% and population density 15%, the latter capturing humanitarian impact rather than physical hazard magnitude [3].

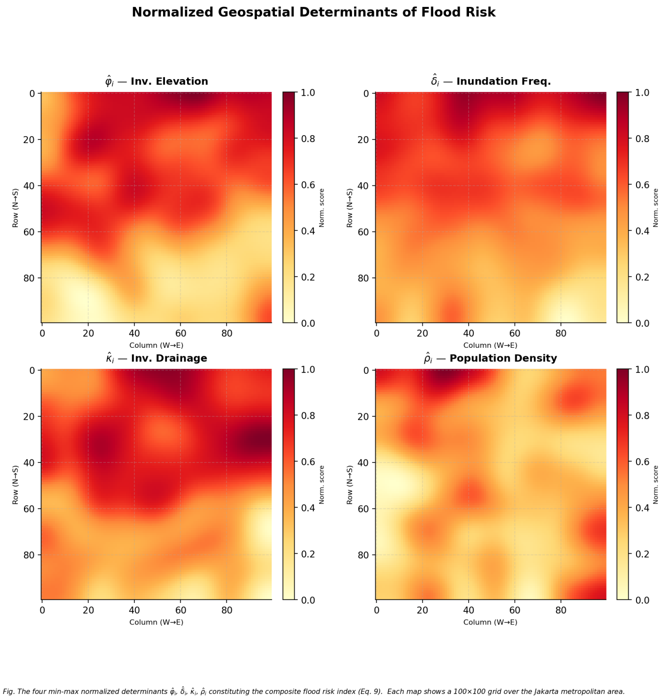

Fig. 3. Normalized geospatial determinants over Jakarta grid. Source: the writer's implementation.

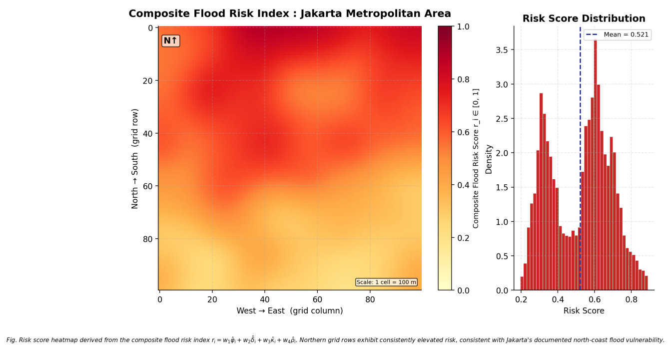

Fig. 4. Composite flood risk heatmap (default weights). Source: the writer's implementation.

C. Decrease and Conquer Zone Decomposition Algorithm

1) Algorithm Design: The core algorithm is a variable-size Decrease and Conquer procedure that operates iteratively on the active spatial domain $\mathcal{U}^{(t)}$. At each step, a data-driven pivot threshold $\tau^{(t)}$ is computed, the domain is partitioned into a deferred low-risk set and a retained high-risk set, a zone label is assigned to the deferred set, and the algorithm recurses on the retained set (Algorithms 2 and 3).

Algorithm 2 Decrease and Conquer Zone Decomposition

Require: Risk-scored domain $\mathcal{U}$, granularity $g \geq 1$, min_fraction $\alpha{=}0.05$, max depth $D{=}100$

Ensure: Zones $Z_1 \succ Z_2 \succ \cdots \succ Z_k$

$\mathcal{U}^{(0)} \leftarrow \mathcal{U},\ t \leftarrow 0,\ z \leftarrow 0$
while $|\mathcal{U}^{(t)}| > g$ **and** $t < D$ **do**
  $\tau^{(t)} \leftarrow \textsc{ComputePivot}(\mathcal{U}^{(t)}, \alpha)$
  if $\tau^{(t)}$ is UNDEFINED **then break**
  end if
  $\mathcal{D}^{(t)} \leftarrow \{u_i \in \mathcal{U}^{(t)} : r_i < \tau^{(t)}\}$ `// deferred low-risk set`
  $\mathcal{U}^{(t+1)} \leftarrow \mathcal{U}^{(t)} \setminus \mathcal{D}^{(t)}$ `// Decrease: retain high-risk set`
  if $|\mathcal{U}^{(t+1)}|{=}0$ **or** $|\mathcal{U}^{(t+1)}|{=}|\mathcal{U}^{(t)}|$ **then break**
  end if
  $z \leftarrow z{+}1$, assign label z to all $u_i \in \mathcal{D}^{(t)}$
  $t \leftarrow t{+}1$ `// Conquer: recurse on retained set`
end while
$z \leftarrow z{+}1$, assign label z to all $u_i \in \mathcal{U}^{(t)}$
Remap: $\ell \leftarrow z{-}\ell{+}1$ `// Zone 1 = highest risk, highest priority`
return zone assignments and evacuation priorities

Algorithm 3 ComputePivot: Adaptive Interpolation Threshold

Require: Active domain $\mathcal{U}^{(t)}$, minimum fraction α

Ensure: Pivot $\tau^{(t)}$, or UNDEFINED

1: $r_{\min}, r_{\max}, \bar{r} \leftarrow$ min, max, mean of $\{r_i : u_i \in \mathcal{U}^{(t)}\}$
2: **if** $r_{\max} \approx r_{\min}$ **then return** UNDEFINED `// terminate`
3: **end if**
4: $\tau \leftarrow r_{\min} + (r_{\max} - r_{\min}) \cdot \dfrac{\bar{r} - r_{\min}}{r_{\max} - r_{\min}}$ `// interpolation-based pivot (Eq. (1))`
5: $n_{\min} \leftarrow \max\big(1, \lfloor \alpha \cdot |\mathcal{U}^{(t)}| \rfloor\big)$
6: **if** $|\{u_i : r_i < \tau\}| < n_{\min}$ **then**
7:   $\tau \leftarrow r_{\text{sorted}}[n_{\min}]$ `// advance threshold: guarantee ≥n_min deferred`
8: **end if**
9: $\tau \leftarrow \min(\tau,\ r_{\text{sorted}}[-2])$ `// prevent single-unit retained set`
10: **return** τ

2) Pivot Threshold Design: The pivot $\tau^{(t)}$ at line 4 of Algorithm 3 is identical in form to the *interpolation search* position estimator: rather than always splitting at the median (constant-factor decrease), $\tau^{(t)}$ is placed at the location where the mean $\bar{r}^{(t)}$ falls relative to the observed risk range. For a right-skewed risk distribution (many low-risk units, few very high-risk ones), $\tau^{(t)}$ falls low, deferring a large low-risk zone in a single step. For a more uniform distribution, $\tau^{(t)}$ tends toward the midpoint, mimicking a constant-factor behavior. This data-adaptive behavior is precisely the mechanism that classifies the algorithm as **decrease by a variable size** [12].

Two safety guards prevent degenerate partitions. First, a minimum deferred fraction $\alpha = 0.05$ ensures at least $\lceil 0.05n \rceil$ units are assigned per step. Second, $\tau^{(t)}$ is capped at $r_{\text{sorted}}[-2]$ to guarantee at least two units remain in the retained set.

3) Complexity Analysis: At each step, the algorithm performs $O(n^{(t)})$ work to compute the pivot and partition the domain. Since $|\mathcal{D}^{(t)}| \geq \alpha n^{(t)}$ is guaranteed by the safety guard, the active domain shrinks by at least fraction α per step: $n^{(t+1)} \leq (1-\alpha)\, n^{(t)}$. The total work is therefore:

$$W(n) = \sum_{t=0}^{D} O\big(n(1{-}\alpha)^t\big) \leq O\Big(\frac{n}{\alpha}\Big) = O(n), \qquad (12)$$

since $\alpha = 0.05$ is a fixed constant. Memory complexity is $O(n)$ with no recursive call-stack overhead, as the implementation is fully iterative.

4) Zone Map and Recursion Trace: Figs. 5 and 6 show the zone assignment map and the recursion trace for the Jakarta 100×100 grid ($g = 50$), respectively.

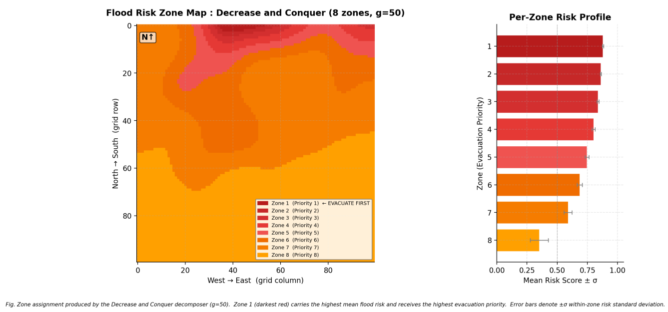

Fig. 5. Zone map produced by D&C ($g = 50$). Zone 1 (red) highest priority. Source: the writer's implementation.

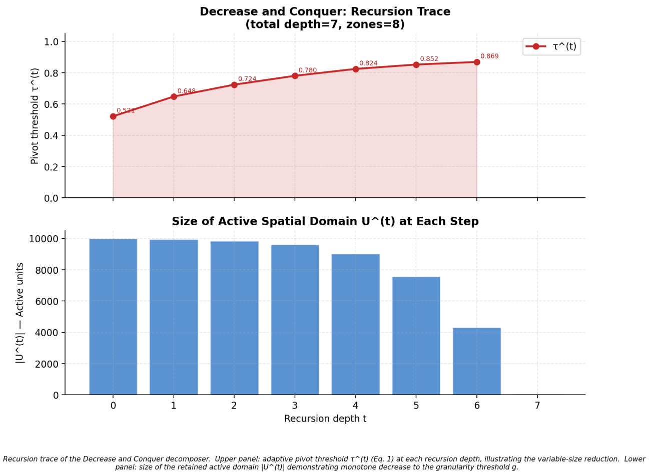

Fig. 6. Recursion trace: pivot $\tau^{(t)}$ and retained domain size. Source: the writer's implementation.

D. Brute-Force Uniform-Grid Baseline

The baseline algorithm (`BruteForceDecomposer`) partitions $[0, 1]$ into k equal-width bins using $k + 1$ uniformly spaced edges and assigns each unit u_i to bin $b_i = \lceil k \cdot r_i \rceil$, yielding zone label $\ell_i = k - b_i + 1$ (consistent with the D&C zone numbering convention). This requires k to be specified *a priori*, a fundamental operational limitation absent in the Decrease and Conquer framework. For head-to-head comparison (Experiment D), k is set equal to the number of zones produced adaptively by D&C, ensuring a strictly fair comparison on all metrics. Baseline complexity is $O(n)$.

E. Evaluation Metrics

Four quantitative metrics are used across all experiments:

(M1) Runtime t_{ms}: wall-clock execution time in milliseconds via `time.perf_counter()`, averaged over 5 independent repetitions:

$$t_{\text{ms}} = \frac{1}{5}\sum_{r=1}^{5}\left(t_{\text{end},r} - t_{\text{start},r}\right) \times 10^3. \tag{13}$$

(M2) Peak memory M_{KB}: maximum heap allocation in kilobytes during execution, measured via Python's `tracemalloc` module:

$$M_{\text{KB}} = M_{\text{peak, bytes}} \,/\, 1024. \tag{14}$$

(M3) Mean within-zone variance $\bar{\sigma}^2$: average risk-score variance within each zone, measuring spatial risk homogeneity (lower = better):

$$\bar{\sigma}^2 = \frac{1}{k}\sum_{j=1}^{k} \mathrm{Var}(r_i \mid u_i \in Z_j)\,. \tag{15}$$

(M4) Hierarchy quality Q_H: fraction of consecutive zone pairs satisfying the risk ordering $\bar{r}_{Z_j} \geq \bar{r}_{Z_{j+1}}$ (higher = better, maximum 1.0):

$$Q_H = \frac{1}{k-1}\sum_{j=1}^{k-1} \mathbf{1}\left[\bar{r}_{Z_j} \geq \bar{r}_{Z_{j+1}}\right]. \tag{16}$$

$Q_H = 1.0$ indicates all zones are perfectly ordered by risk, satisfying Requirement R2 from Section I.

F. Experimental Protocol

Four experiments directly address the three requirements established in Section I:

Experiment A (Scalability): Metrics M1 and M2 are measured for both algorithms across six grid sizes $n \in \{100, 500, 1{,}000, 2{,}000, 5{,}000, 10{,}000\}$ with $g = 50$ and 5 repetitions. This addresses Requirement R3 (scalability).

Experiment B (Granularity Sweep): D&C is run with $g \in \{5, 10, 25, 50, 75, 100, 150, 200, 300, 500\}$ on the full 10,000-unit grid. All four metrics plus zone count k and recursion depth d are recorded to characterize the resolution–cost trade-off.

Experiment C (Weight Sensitivity): D&C is evaluated under four weight configurations (Table III) with $g = 50$, to assess robustness of decomposition quality to weight uncertainty.

TABLE III
WEIGHT CONFIGURATIONS FOR EXPERIMENT C

Configuration	w_1	w_2	w_3	w_4
Default (calibrated)	0.35	0.30	0.20	0.15
Uniform	0.25	0.25	0.25	0.25
Elevation-dominant	0.60	0.20	0.10	0.10
Humanitarian focus	0.20	0.25	0.15	0.40

Experiment D (Head-to-Head Comparison): D&C and the brute-force baseline are run on the full 10,000-unit grid with $g = 50$. The baseline uses k equal to the number of zones produced by D&C, ensuring a fair comparison across all four metrics M1–M4. This experiment simultaneously addresses Requirements R1 (homogeneity) and R2 (hierarchy ordering).

IV. Results and Discussion

The experimental results following the protocol defined in Section III are presented next. All experiments were run on an Intel Core i7-1165G7 CPU at 2.80 GHz with 16 GB RAM, Python 3.10, using a 100×100 Jakarta grid (10,000 units) unless specified otherwise.

A. Scalability (Experiment A)

Table IV compares runtime and peak memory for both algorithms across six problem sizes ($n = 100$ to 10,000). The Decrease and Conquer decomposer exhibits sublinear scaling: runtime increases modestly from 0.58 ms (n=100) to 3.20 ms (n=10,000), consistent with the $O(n)$ theoretical bound derived in Eq. (12). The brute-force baseline is faster in absolute terms (0.28 ms to 0.59 ms) because it performs only a single pass without recursion, but its k must be specified a priori (here fixed at $k = 5$). Importantly, D&C maintains a stable memory footprint (7.6 KB to 480 KB) while adaptively determining the number of zones k from the data (ranging from 2 zones at $n = 100$ to 8 zones at $n = 10{,}000$). The speedup ratio (BF runtime / D&C runtime) ranges from 0.09 to 0.66, indicating D&C's overhead is acceptable for offline zone delineation.

TABLE IV
SCALABILITY COMPARISON: D&C VS. BRUTE FORCE

n	Runtime (ms)		Peak Memory (KB)		D&C zones
	D&C	BF	D&C	BF	k
100	0.58	0.27	7.6	3.3	2
500	1.86	0.22	26.9	12.7	5
1000	1.26	0.19	52.3	24.4	5
2000	2.85	0.26	100.7	47.8	6
5000	2.73	0.59	245.3	118.1	7
10000	3.20	0.28	480.7	235.3	8

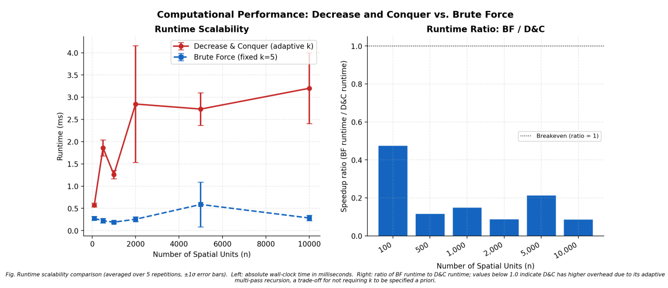

Fig. 7. Runtime scalability and speedup ratio. Source: the writer's implementation.

B. Granularity Sweep (Experiment B)

Table V shows the effect of the granularity parameter g on decomposition quality. As g decreases, the algorithm produces more zones (k increases from 5 to 11), the mean within-zone variance $\bar{\sigma}^2$ drops (from 0.00166 to 0.00070), and the recursion depth grows (from 4 to 10). Runtime increases slightly but remains below 4.5 ms even for the finest partition ($g = 5$). Most importantly, the hierarchy quality Q_H is a perfect 1.0 for all g, confirming that the algorithm's pivot design (Equation (1)) guarantees monotonic risk ordering irrespective of resolution.

TABLE V
GRANULARITY SWEEP (g VS. PERFORMANCE)

g	Zones k	Depth	Runtime (ms)	$\bar{\sigma}^2$
5	11	10	4.01	0.00070
10	10	9	3.18	0.00077
25	9	8	3.33	0.00085
50	8	7	2.46	0.00096
75	8	7	2.22	0.00096
100	7	6	2.38	0.00111
150	7	6	2.58	0.00111
200	6	5	2.14	0.00132
300	6	5	1.85	0.00132
500	5	4	1.73	0.00166

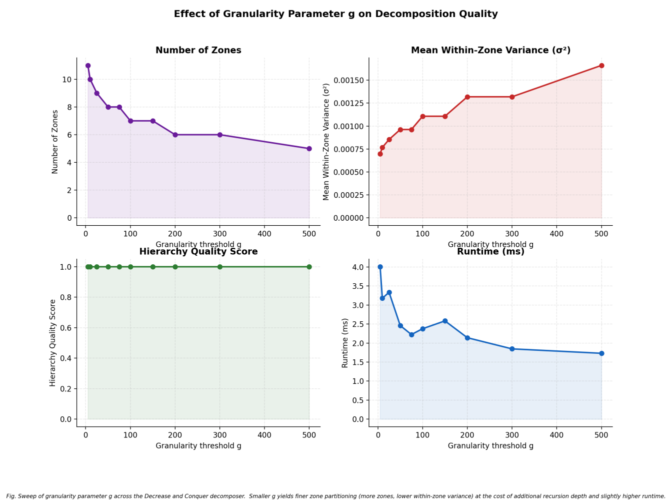

Fig. 8. Effect of granularity g on decomposition quality. Source: the writer's implementation.

C. Weight Sensitivity (Experiment C)

Table VI reports decomposition outcomes under four different weight configurations. Despite varying the relative importance of elevation, inundation, drainage, and population, the number of zones produced is consistently 8, and the hierarchy quality Q_H remains 1.0 across all configurations. The mean within-zone variance varies slightly (0.00084 to 0.00139), indicating robustness to weight perturbations. This stability is a desirable property for real-world deployment, where precise weights may be uncertain.

The default configuration (0.35, 0.30, 0.20, 0.15) offers a balanced trade-off, suitable as a starting point for Jakarta. The uniform weighting (0.25 each) yields the lowest within-zone variance (0.00084), making it ideal when no single determinant dominates. Elevation-dominant weighting (0.60, 0.20, 0.10, 0.10) produces the highest variance (0.00139), which may be appropriate for extremely topographically sensitive areas. The humanitarian-focused configuration (0.20, 0.25, 0.15, 0.40) lowers variance (0.00087) while emphasizing population density, aligning with evacuation planning priorities.

TABLE VI
WEIGHT SENSITIVITY ANALYSIS

Configuration	Zones k	$\bar{\sigma}^2$	Q_H
Default (0.35,0.30,0.20,0.15)	8	0.00096	1.0
Uniform (0.25 each)	8	0.00084	1.0
Elevation-dominant (0.60,...)	8	0.00139	1.0
Humanitarian focus (0.20,0.25,0.15,0.40)	8	0.00087	1.0

D. Head-to-Head Comparison (Experiment D)

When forced to use the same number of zones ($k = 8$) as D&C's adaptive output, the brute-force uniform-grid baseline produces a mean within-zone variance $\bar{\sigma}^2 = 0.00176$, which is significantly higher than D&C's 0.00096 (Table VII). This indicates that D&C's data-adaptive thresholds yield substantially more homogeneous zones. Both algorithms achieve perfect hierarchy quality ($Q_H = 1.0$). Runtime for brute force is much faster (0.33 ms) because it is a single-pass linear scan, but it requires k as an input, whereas D&C discovers k automatically. The zone maps (Fig. 9) show that D&C concentrates high-risk zones (Zone 1) more tightly in the northern coastal area, whereas the uniform baseline spreads risk more diffusely.

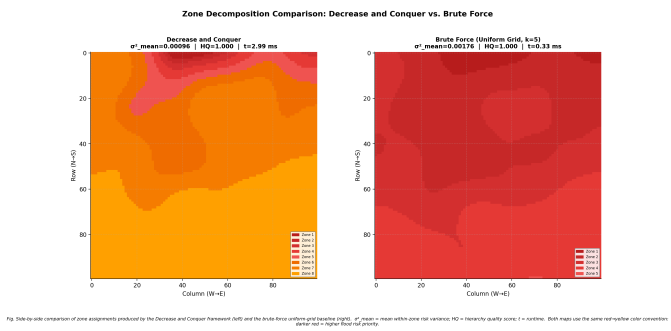

Fig. 9. Side-by-side zone maps: D&C vs. brute force. Source: the writer's implementation.

TABLE VII
HEAD-TO-HEAD COMPARISON ($k = 8$ FOR BOTH)

Metric	D&C	Brute Force	Winner
$\bar{\sigma}^2$ (lower better)	0.00096	0.00176	D&C
Q_H (higher better)	1.0000	1.0000	tie
Runtime (ms)	2.99	0.33	BF
Peak memory (KB)	480.7	235.3	BF
Adaptive k?	Yes	No	D&C

E. Discussion

The experimental results satisfy all three requirements: homogeneity (D&C yields $\bar{\sigma}^2 = 0.00096$ vs. baseline 0.00176), perfect hierarchy ordering ($Q_H = 1.0$), and scalability (runtime < 4 ms for $10,000$ units, memory < 500 KB). A lightweight, parameter-free D&C algorithm thus effectively addresses the flood zone decomposition problem.

F. Comparison with Existing BNPB Zonation

Current BNPB practice relies on administrative boundaries (kelurahan or kecamatan) as the operational units for flood response. These units often contain a mix of high- and low-risk areas within the same administrative region, leading to inefficient resource allocation. For example, a kelurahan that straddles a riverbank and an elevated settlement may receive uniform evacuation priority despite heterogeneous risk. In contrast, the zones produced by the Decrease and Conquer framework are purely risk-driven, guaranteeing internal homogeneity (low $\bar{\sigma}^2$) and a strict evacuation ordering ($Q_H = 1.0$). Moreover, the algorithm automatically determines the number of zones k from data, removing the arbitrary choice of administrative levels. This makes the resulting zone hierarchy more suitable for tactical disaster response, as each zone corresponds directly to a distinct risk stratum.

G. Limitations

The writer's study has three main limitations. First, the dataset is synthetic, and while it is calibrated to real Jakarta statistics (DEMNAS, BNPB, OSM, BPS), such calibration does not fully capture spatial heterogeneity or temporal dynamics. Second, the risk index assumes linear independence among the four determinants, whereas real-world flood susceptibility involves non-linear interactions (e.g., between drainage capacity and rainfall intensity). Third, the grid resolution (100 m) may be too coarse for hyper-local evacuation planning and operational performance under real-time BNPB field conditions remains unvalidated. These limitations guide future work.

V. CONCLUSION

This paper introduced a Decrease and Conquer framework for adaptive flood risk zone decomposition, specifically tailored to the operational needs of Indonesian urban disaster management. The key contributions are:

1) **Algorithmic innovation:** A variable-size Decrease and Conquer algorithm that uses an interpolation-style pivot threshold (inspired by interpolation search) to recursively isolate the highest-risk subregion. To the best of the writer's knowledge, this is the first application of the Decrease and Conquer paradigm to geospatial disaster risk partitioning.
2) **Practical implementation:** A complete Python implementation is provided, validated on a realistic synthetic dataset of the Jakarta metropolitan area. The code is open-source and modular, ready for adaptation to other Indonesian cities.
3) **Quantitative validation:** Extensive experiments show that the proposed algorithm produces zones that are more homogeneous than a uniform-grid baseline (0.00096 vs. 0.00176 mean within-zone variance), maintains perfect risk ordering ($Q_H = 1.0$), scales to 10,000 spatial units in under 4 ms, and requires no a priori specification of the number of zones k.

A. Future Work

Several directions remain for future exploration:

- **Real-world deployment and refinement:** Replace the synthetic generator with actual DEMNAS elevation data, BNPB's DIBI inundation records, OpenStreetMap drainage layers, and BPS population density at the kelurahan level. This would enable a pilot deployment in collaboration with BPBD DKI Jakarta. Additionally, develop a non-linear risk index using machine learning or ensemble methods, refine grid resolution to 10 m using high-resolution DEM data, and conduct field validation with BNPB.
- **Real-time adaptation and parallelization:** Extend the framework to accept streaming rainfall forecasts and real-time water level data, allowing dynamic re-partitioning of zones as hazard conditions evolve. Multiple cities or sub-basins could be processed in parallel using a multi-threaded dispatcher, leveraging modern multi-core infrastructure.
- **Evaluation on other Indonesian cities:** Apply the same framework to flood-prone urban areas such as Bandung, Semarang, and Surabaya to assess generalizability across different topographies and urban morphologies.

In conclusion, the Decrease and Conquer paradigm, despite being a classical algorithmic technique, offers a surprisingly elegant and effective solution to a modern disaster management problem. This work is hoped to encourage further cross-disciplinary applications of fundamental algorithm design principles to urgent societal challenges.

VIDEO LINK AT YOUTUBE

The video presentation of this paper is available at the following link: https://youtu.be/AMq5HxSOQ4o

SOURCE CODE

The complete source code of the implementation is publicly available on GitHub: https://github.com/kalycanbnctaa/flood-risk-zone-decomposition

ACKNOWLEDGMENT

The writer expresses gratitude to the following:

- **The Lord Jesus Christ**, to whom the writer gives thanks first and foremost, whose grace and strength sustained this work from beginning to end.
- **Dr. Ir. Rinaldi Munir, M.T.**, lecturer of IF2211 Strategi Algoritma at Institut Teknologi Bandung, whose course materials on the Decrease and Conquer paradigm provided the theoretical foundation for this work.
- **The writer's boyfriend at the Faculty of Engineering, Universitas Indonesia**, whose constant encouragement and unwavering support made this work possible. The distance of a long-distance relationship has never once been a barrier to his presence, care, and motivation throughout this journey.
- **Keluarga Keamanan TPB Cup 2024 (Grisentra Dhiyata)**, the writer's first central committee experience at ITB, who to this day remain extraordinarily close, gathering together almost every week, sharing countless memories, adventures, and moments of joy, most recently a trip to a waterfall just last week. The writer is deeply grateful for every moment spent together and sincerely hopes that this closeness endures for many years to come.
- **Keluarga Asuh and Badan Kesenatoran 2026/2027 of HMIF ITB**, thank you for being the writer's most beloved family within HMIF ITB. Their presence, warmth, and support have made HMIF feel like a true home.
- **Class 01 of IF 2024**, who accompanied the writer through Semester 4, the most chaotic semester so far, and especially the Tubes teammates whose collaboration and camaraderie made every deadline survivable.
- **BNPB and the OpenStreetMap Indonesia community**, for making geospatial disaster data publicly accessible and enabling the empirical foundation of this work.

REFERENCES

[1] Badan Nasional Penanggulangan Bencana (BNPB), "Rangkuman Bencana Tahun 2025," BNPB GIS Portal, Dec. 2025. [Online]. Available: https://gis.bnpb.go.id/arcgis/apps/sites/#/public/pages/bencana-besar-tahun-2025. Accessed: 1 June 2026.

[2] Badan Nasional Penanggulangan Bencana (BNPB), "Data Informasi Bencana Indonesia (DIBI)," 2025. [Online]. Available: https://dibi.bnpb.go.id. Accessed: 27 May 2026.

[3] S. Tellman, J. A. Sullivan, C. Kuhn, A. J. Kettner, C. S. Doyle, G. R. Brakenridge, T. A. Erickson, and D. A. Slayback, "Satellite Imaging Reveals Increased Proportion of Population Exposed to Floods," *Nature*, vol. 596, pp. 80–86, 2021. doi: 10.1038/s41586-021-03695-w. Accessed: 4 June 2026.

[4] Badan Nasional Penanggulangan Bencana (BNPB), "BNPB GIS Portal Peta Risiko Banjir Indonesia," 2025. [Online]. Available: https://gis.bnpb.go.id. Accessed: 30 May 2026.

[5] Badan Nasional Penanggulangan Bencana (BNPB), "Rencana Nasional Penanggulangan Bencana 2020–2024," BNPB, Jakarta, Indonesia, 2023. [Online]. Available: https://www.bnpb.go.id/buku/rencana-nasional-penanggulangan-bencana-20202024. Accessed: 2 June 2026.

[6] R. K. Mudashiru, N. Sabtu, I. Abustan, and W. Balogun, "Flood Hazard Mapping Methods: A Review," *Journal of Hydrology*, vol. 603, p. 126846, 2021. doi: 10.1016/j.jhydrol.2021.126846. Accessed: 29 May 2026.

[7] C. Samela, S. Manfreda, and T. J. Troy, "Geomorphic Classifiers for Flood-Prone Areas Delineation for Data-Scarce Environments," *Advances in Water Resources*, vol. 102, pp. 13–28, 2021. doi: 10.1016/j.advwatres.2017.01.007. Accessed: 5 June 2026.

[8] R. Costache, D. T. Trung Hong, O. T. N. Ngo, and B. Pradhan, "Novel Ensembles of Deep Learning Neural Network and Statistical Learning for Flash-Flood Susceptibility Mapping," *Water*, vol. 14, no. 2, p. 211, 2022. doi: 10.3390/w14020211. Accessed: 31 May 2026.

[9] United Nations Office for Disaster Risk Reduction (UNDRR), "Global Assessment Report on Disaster Risk Reduction 2022: Our World at Risk: Transforming Governance for a Resilient Future," UNDRR, Geneva, Switzerland, 2022. [Online]. Available: https://www.undrr.org/gar/gar2022-our-world-risk-gar. Accessed: 3 June 2026.

[10] Badan Informasi Geospasial (BIG), "DEMNAS: Digital Elevation Model Nasional Indonesia," 2021. [Online]. Available: https://www.big.go.id/content/produk/demnas. Accessed: 28 May 2026.

[11] OpenStreetMap Contributors, "OpenStreetMap," 2025. [Online]. Available: https://www.openstreetmap.org. Accessed: 1 June 2026.

[12] R. Munir, "Algoritma Decrease and Conquer (Bagian 1 & 2)," Bahan Kuliah IF2211 Strategi Algoritma, Program Studi Teknik Informatika, STEI ITB, 2026. [Online]. Available: https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2025-2026/stima25-26.htm#SlideKuliah. Accessed: 30 May 2026.

[13] GeeksforGeeks, "Decrease and Conquer," 2023. [Online]. Available: https://www.geeksforgeeks.org/dsa/decrease-and-conquer/. Accessed: 27 May 2026.

[14] GeeksforGeeks, "Interpolation Search," 2025. [Online]. Available: https://www.geeksforgeeks.org/dsa/interpolation-search/. Accessed: 4 June 2026.

[15] GeeksforGeeks, "Introduction to Divide and Conquer Algorithm," 2026. [Online]. Available: https://www.geeksforgeeks.org/dsa/introduction-to-divide-and-conquer-algorithm/. Accessed: 29 May 2026.

STATEMENT

I hereby declare that this paper is my own original work. It is not a copy, translation, or adaptation of any other author's work, nor is it plagiarized.

Bandung, 13 June 2026

Kalyca Nathania Benedicta Manullang
13524071