Issues in Information Systems
Volume 25, Issue 2, pp. 214-221, 2024

DOI: https://doi.org/10.48009/2_iis_2024_117

Decision support system for post-disaster resilience

Baba Nasiru Junior Lawal, *Texas A & M University Kingsville, nasirujunior77@gmail.com*
Joon-Yeoul Oh, *Texas A & M University Kingsville, joon-yeoul.Oh@tamuk.edu*
David Hicks, *Texas A & M University Kingsville, david.hicks@tamuk.edu*
Vinod Kumar, *Texas A & M University Kingsville, vinod.kumar@tamuk.edu*

Abstract

Efficient resource allocation after a disaster is critical to enhance the resilience of social infrastructures. This research proposes a post-disaster decision-support system (PDDSS) to make an effective yet efficient decision in timely manner for allocating the resources after the disasters. The location-based PDDSS utilizes the integration of multiple sources of data with a modified Analytical Hierarchical Process (AHP) to enhance the effectiveness of the decision. The various test cases validate the feasibility of the Python programming based PDDSS and demonstrate that the proposed system facilitates disaster recovery to enhance the effectiveness of resource distribution in the post-disaster scenarios.

Keywords: decision support system, resilience, disaster management, resource allocation, scheduling, python programming

Introduction

The decision Support System (DSS) has become a vital tool in various fields, allowing decision-makers to analyze vast datasets and make informed decisions promptly and effectively (Phillips-Wren, Daly, & Burstein, 2021). Typical natural disasters, such as floods, hurricanes, earthquakes, and pandemics disrupt societies and economies, and the societies need a post-disaster management for an effective resilience. After the disasters passed, one of the most important parts is to allocate the resources, such as labor, budget, and equipment, for the necessary sections so that citizens' lives can get back to normal quickly.

However, there are some challenging considerations for distributing the resources efficiently and equitably because such resources are limited (Hinojosa, 2023). The objective of this research is to propose a framework for a real-time post-disaster decision support system (PDDSS) to make an effective yet efficient decision in a timely manner for allocating the resources after the disasters. To achieve the objective of this research, some test cases are developed and tested.

A Python program is used to utilize the system for analyzing historical data, adapting to current circumstances, and enhancing the effectiveness of the decision. In general, the resource allocation problem is multi-criteria decision making with a large number of alternatives (Hessami, 2018). The system needs to integrate data for accuracy from the diverse sources, such as resource availability, geographic location, damage extent, and specific needs of the affected population. Due to the limitation of the resources, the

large number of alternatives should be prioritized based on some factors, such as the severity of the damages, the number of citizens affected and size of the damages, etc.
A sustainable and efficient resource allocation in emergencies guarantees smooth operations and rational resource sharing (Wang, 2021). Huang and Rafiei (2019) emphasized efficiency and equity as the foundational principles in emergency resource allocation. Based on efficiency, items are prioritized for the cost-effective resource delivery, while equity ensures fair distribution, especially in humanitarian relief efforts (Huang & Rafiei, 2019). For data integration, risk assessment, and resource allocation capabilities, the DSS should integrate with diverse software applications, analytical models, GIS, simulation models, and machine learning algorithms (Franco, 2016).

During the COVID-19 pandemic, Moulaei et al. (2022) developed the clinical decision support systems (CDSSs) and successfully managed the various applications repeatedly, while providing quality decisions, eliminating unnecessary testing, increasing patient safety, and potentially reducing harmful and dangerous complications. Including Moulaei et al.'s (2022) work, most of the previous researches have dwelled on public health and safety incidents and not many researches covered for post-disaster resiliencies.

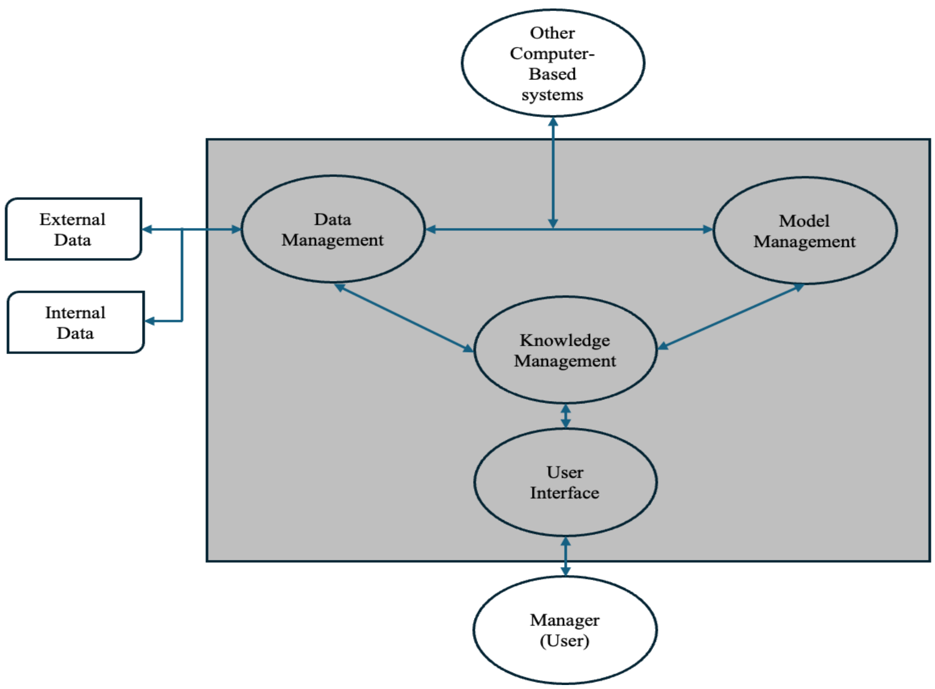

Figure 1: Components of PDDSS

Disasters frequently occur globally, so systematic support is needed for integrating diverse disciplines, such as engineering, environmental science, and economics, etc., allowing for the identification of potential risks such as property damage, population displacement, and economic disruption (Rye & Aktas, 2022). An effective mitigation strategy, including efficient recovery systems, can be formulated to address the challenges, such as the integration of multiple sources of data, data quality and stakeholder engagement (Sabbaghtorkan et al, 2020).

The key factor in most DSS is the data integration for assessment and resource allocation. Even though many researchers have developed decision support systems to tackle these issues, such as integrating data from heterogenous sources with different formats, further research is needed to fully capture the complexity of the disaster management system and provide comprehensive decision support for resilience. The

Analytical Hierarchical Process (AHP) technique is frequently used for multi criteria decision making. The technique requires one to construct a hierarchy of criteria and sub criteria, allocating weights to criteria, assigning numerical values to evaluation dimensions, and finally summing the products of numeric evaluations and weights (Pinto, 2021).

Literature Review

Several researchers have developed and tested PDDSSs to facilitate efficient resource allocation. Ahmadi et al. (2022) presented a novel decision support model for allocating and routing search and rescue (SAR) resources in post-earthquake districts. A two-phase decomposition methodology is employed to structure the resource allocation problem as iterative interconnected phases of mixed-integer programming (MIP) models.

Cavdur et al. (2020) presented a spreadsheet-based decision support system for post-disaster facilities allocation. The tool comprises three main components: a user interface, a database, and a decision engine. The user interface is developed using Microsoft Excel, the database is implemented on the Microsoft SQL server, and the decision engine is implemented using the Gurobi optimizer and the Maximal Software Mathematical Programming Language (MPL)-OptiMax.

Song, Jun, and Lee (2024) proposed a multi-facility, multi-period, and bi-objective optimization approach to PDDSS. A bi-objective mathematical is implemented to facilitate multi-period disaster response and recovery by efficiently allocating resources, including warehouses, emergency medical facilities, shelters, and drone stations. The results indicated the PDDSS's ability to offer effective operational plans and guidelines to solve the bi-objective optimization problem in disaster response and recovery systems

Kant, Machavarapu, and Natha (2023) constructed a PDDSS based on the Fuzzy Analytic Hierarchy Process (FAHP) and the Best Worst Method (BWM). The FAHP is calculated manually and modelled from an Excel spreadsheet using MATLAB. Conversely, the BWM method is used to compare multicriteria decision-making strategies manually. Comparisons of the two approaches indicate similar feasibility for resolving supply chain management problems after disasters. The reviewed previous works collectively indicate the possibility of basing PDDSS on different methodologies and concepts. Some of the leading concepts include MIP models, BWM, AHP, FAHP, and spreadsheets. Resource allocation for a post-disaster recovery is still an active research area.

Methodology

The major focus of this research methodology is to develop a framework for resource allocation and distribution using a location-based service with Python programming. For the accurate distribution of resources, factors such as the resources available, the amount of highly demanded resources, the period taken to deploy the resources, and the ability to restore operations, should be carefully considered.

The structure of the decision support system for post-disaster reflects the process of risk management because the incidents resulting from a disaster are considered as the risk items. Figure 2 outlines the flow of information within the post-disaster decision support system, initiating with the establishment of the decision context for the disaster scenario.

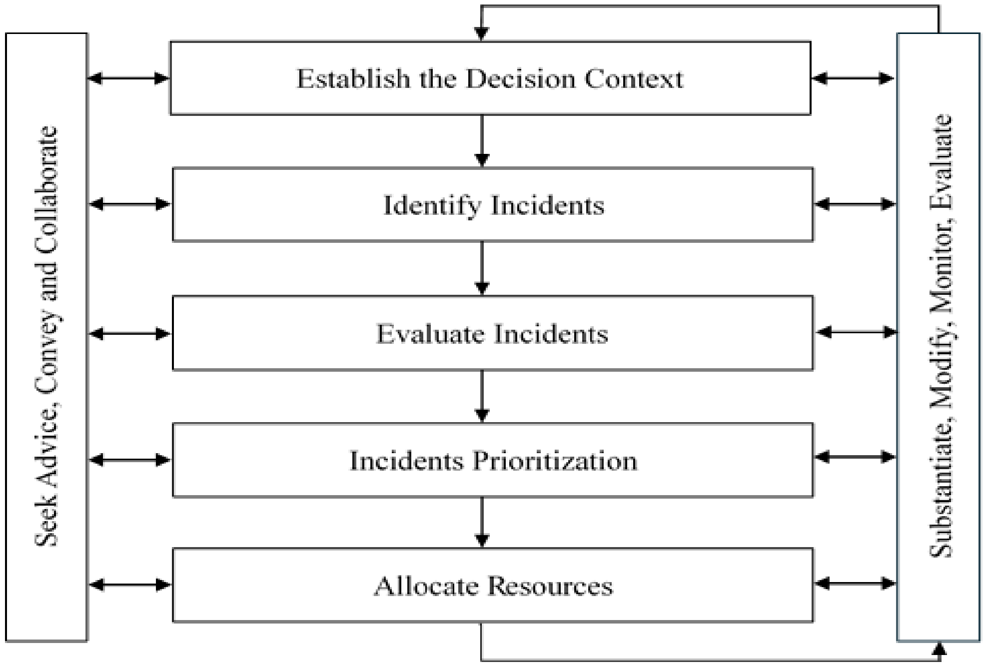

Figure 2: PDDSS Flowchart

Subsequently, it involves identifying the incidents, evaluating them, and ultimately prioritizing them based on their severity and impact. For the post-disaster DSS, a collection of information is needed as user inputs for identifying the incidents or tasks to be performed and collecting the relevant data into the system for evaluating the incidents. The incidents may include a fallen power line obstructing road, dangling road signals, power outages, and damaged water lines. Table 1 shows the examples of user inputs.

Table 1: Example of the user inputs for PDDSS

S/N	User Input
1.	Enter address or allow application to find your location.
2.	Upload picture of damage scene.
3.	Describe the incident.
4.	Enter the estimated number of people affected.
5.	Enter the estimated time required for resilience.
6.	Enter the estimated damage size.
7.	Enter the severity of damage.

The initial phase of the PDDSS involves gathering inputs related to the incidents or potential occurrences, such as location details, visual documentation of damage, affected population estimation, damage specifics, associated tasks, and severity levels. The system needs to provide a user-friendly interface for easy data input using a user input device or cellphone application. Once the necessary information is collected, the incidents should be evaluated for the incident prioritization. The AHP is used to evaluate and prioritize the incidents with the predefined scale of estimation.

The proposed PDDSS was tested with various test scenarios. All scripts are hosted in a GitHub repository, located at https://github.com/GeneralNas/dms_main. Each scenario had a different number of incidents with a different scale of estimation. The pseudocode for the proposed PDDSS is described in Figure 3.

```
// User input file path
User input for file path and transfer LoadExcel(file_path) into a variable df
// Define constants
Set total_hours_per_day; Set daily_hours_per_worker; Set total_number_of_workers;
Set start_date =    CurrentDateTime()
// Calculate and sort required hours for each task using variable df
df['Required Hours'] = df['RESIL TIM(Hrs)'] * df['PEP RQ']
// Initialize tracking structures
Initialize tasks_daily_hours as a dictionary to track daily hours per task
Initialize tasks_start_end_dates as a dictionary to track start and end dates per task
Initialize daily_totals as a dictionary to track total daily hours
// Allocate hours for all tasks
Call AllocateTaskHours
// Prepare final tasks data
tasks_per_day_df = Convert tasks_daily_hours to DataFrame
Filter tasks_per_day_df to include only days with work or start/end dates
// Map start and end dates from timestamps to formatted strings
// Reset index for merging
tasks_per_day_df_reset = ResetIndex(tasks_per_day_df, 'INSTANCES')
// Merge processed task data with sorted DataFrame
merged_df = MergeDataFrames(df_sorted, tasks_per_day_df_reset, on 'INSTANCES')
// Display and save the merged DataFrame
// Functions invoked in the above implementations
Function AllocateTaskHours
Function FormatDate9
```

Figure 3: Pseudocode for PDDSS

It is initiated with the user inputs. For the test cases, the input data was arranged in Excel format. The criteria to evaluate the incidents for prioritization is predefined as described in Table 2. The weights for the criteria are assessed by the comparison between the criteria, which leads to the creation of a square matrix. After the normalization process, the weight is assigned to each criterion. Similarly, these steps are followed to prioritize the incidents for each criterion. This incident prioritization process is crucial for determining resource allocation due to the limitation of resources, so the resources are allocated to the incidents based on the prioritized order.

Table 2: Scale of the criteria estimation for each incident

Scale	Number of people affected	Damaged Area (m^2)	Time required	Severity Levels
1	≤ 10	≤ 1	≤ 1Hr	Minimum level
2	≤ 50	≤ 5	≤ 2Hrs	Low level 1
3	≤ 100	≤ 10	≤ 4Hrs	Low level 2
4	≤ 300	≤ 20	≤ 1 Day	Medium level 1
5	≤ 500	≤ 50	≤ 3 Days	Medium level 2
6	≤ 1000	≤ 100	≤ 1 Week	Medium level 3
7	≤ 3000	≤ 500	≤ 2 Weeks	High level 1
8	≤ 5000	≤ 1,000	≤ 3 Weeks	High Level 2
9	≤ 10,000	≤ 10,000	≤ 1 Month	High Level 3
10	More than 10,000	More than 10,000	More than a month	Maximum level

For the initial resource allocation to the prioritized incidents, the concept of resource loading is used. For the test cases with resource overloading, in which the initial resource allocation exceeds the resource limits at some periods, the resource levelling task is performed to provide not only reducing the overall work time but also enhancing efficiency. The PDDSS framework enables the command line interface (CLI) to capture user location and input resilience parameters, such as resilience time and required personnel count. Notably, the scripts are meticulously annotated for clarity. Table 3 illustrates an example of how the output is presented.

Table 3: PDDSS Output example

TASK	RANK	RES. TIME (HRS)	PEOPLE REQUIRED	HOURS NEEDED	D 1	D 2	D 3	D 4	D 5	D 6	START	END	DURATION
TASK 28	1	9	2	18	16	2	0	0	0	0	07-May	08-May	2
TASK 30	2	4	3	12	12	0	0	0	0	0	07-May	07-May	1
TASK 14	3	4	6	24	24	0	0	0	0	0	07-May	07-May	1
TASK 8	4	4	2	8	8	0	0	0	0	0	07-May	07-May	1
TASK 5	5	1	5	5	5	0	0	0	0	0	07-May	07-May	1
TASK 20	6	15	4	60	32	28	0	0	0	0	07-May	08-May	2
TASK 12	7	4	7	28	28	0	0	0	0	0	07-May	07-May	1
TASK 27	8	6	10	60	60	0	0	0	0	0	07-May	07-May	1
TASK 3	9	5	6	30	15	15	0	0	0	0	07-May	08-May	2
TASK 29	10	5	3	15	0	15	0	0	0	0	08-May	08-May	1

The proposed PDDSS can help improve real-life disaster management and resilience efforts. For example, the system could have been used to optimize air support and coordination efforts, some of the biggest constraints of the 2024 Panhandle fire (Carver, 2024). The need for air support and other critical response resources could have been prioritized based on variables, such as the geographical range of the fires. The coordination between rescue teams and facilities could have also been optimized to ensure effective response and minimal resource wastage.

The PDDSS also marks a significant step forward in disaster management practices. By integrating advanced technologies, data analysis, and decision algorithms, the PDDSS offers invaluable assistance to decision-makers in efficiently distributing resources, prioritizing tasks, and coordinating response efforts. Moreover, the tool enhances situational awareness, facilitates quick decision-making, and optimizes resource usage after disasters. The adaptable nature of the PDDSS also allows for continuous refinement, ensuring its effectiveness in evolving disaster scenarios and changing environmental conditions.

Discussion and Conclusion

Resilience enables a community to rebound from these adverse events and emerges as an essential guiding principle for a post-disaster management. Through comprehensive preparedness, response, and recovery strategies, resilience empowers communities to adapt and thrive when facing adversity. The development of a PDDSS, location-based Python program, for resilience represents a significant advancement in disaster management. By integrating advanced technologies and decision algorithms, the PDDSS assists the decision-makers efficiently for distributing the resources and coordinating the recovery process. The proposed PDDSS enhances the situational awareness, facilitates quick resilience decision-making, and optimizes resource usage for fostering resilient communities and mitigating the impacts from future disasters. A frontend CLI, linked with the modified AHP algorithm in the backend, aids the decision-makers in allocating resources effectively. The successful implementation of backend scripting using Python programming offers advantages in speed, flexibility, and scalability over another tool, such as MATLAB.

Even though the proposed PDDSS performs adequately, there are several limitations, which should be addressed for future work. This research only scripted the PDDSS test scenarios in Python and hosted them on GitHub rather than developing an outright frontend application for direct use. Thus, future research should be expanded for developing a frontend application and enabling seamless backend integration, facilitating easy deployment, scalability, social coding (GitHub), and eventual commercialization. A simple and fitting user interface can be developed using Excel, as illustrated by Cavdur et al. (2020). This research had a limited cover on the data integration and scalability challenges of the PDDSS, which should also be addressed in future study.

References

Ahmadi, G., Tavakkoli-Moghaddam, R., Baboli, A., & Najafi, M. (2022). A decision support model for robust allocation and routing of search and rescue resources after earthquake: A case study. *Operational Research*, 1-43. https://doi.org/10.1007/s12351-020-00591-5

Carver, J. L. (2024, May 1). Decayed power pole sparked the largest wildfire in state history, Texas House committee confirms. *Texas Tribune*. https://www.texastribune.org/2024/05/01/texas-panhandle-wildfires-report/

Cioca, M., & Cioc, L. (2010). Decision support systems used in disaster management. *IntechOpen*, 371-390. https://doi.org/10.5772/39452

Franco, M. J. (2016). Post Disaster Assessment with Decision Support System. *Asia Pacific Journal of Multidisciplinary Research*, 4(2), 140-147. https://www.academia.edu/81068344/Post_Disaster_Assessment_with_Decision_Support_System

Hessami, A., Oh, J.-Y. (2018). Multicriteria Decision-making in Situations with a Large Number of Alternatives, Proceedings of the International Conference on Industrial Engineering and Operations Management, Washington DC, USA, September 27–29. https://ieomsociety.org/dc2018/global-engineering-education/

Hinojosa, A., Oh, J.-Y. (2023). An Optimization Technique for Ranch Management. *Journal of Applied Business and Economics*, 25(2), 252-260. https://doi.org/10.33423/jabe.v25i2.6111

Huang, K., & Rafiei, R. (2019). Equitable last mile distribution in emergency response. *Computers & Industrial Engineering*, 127(1), 887-900. https://doi.org/10.1016/j.cie.2018.11.025

Jung, D., Tuan, T. V., Quoc Tran, D., Park, M., & Park, S. (2020). Conceptual framework of an intelligent decision support system for smart city disaster management. *Applied Sciences*, 10(2), 666. https://doi.org/10.3390/app10020666

Kant, P., Machavarapu, P. K., & Natha, A. R. (2023). Evaluation of decision support system for disaster management using multicriteria decision techniques: a case study of Alappuzha, Kerala. *Urban Planning and Transport Research*, 11(1), 2262546. https://doi.org/10.1080/21650020.2023.2262546

Kant, P., Machavarapu, P. K., & Natha, A. R. (2023). Evaluation of decision support system for disaster management using multicriteria decision techniques: A case study of Alappuzha, Kerala. *Urban, Planning and Transport Research*, *11*(1), 2262546. https://doi.org/10.1080/21650020.2023.2262546

Moulaei, K. (2022). Diagnosing, managing, and controlling COVID-19 using clinical decision support systems: A study to introduce CDSS applications. *Journal of Biomedical Physics and Engineering*, 12(2), 213-224. https://doi.org/10.31661/jbpe.v0i0.2105-1336

Phillips-Wren, G., Daly, M., & Burstein, F. (2021). Reconciling business intelligence, analytics and decision support systems: More data, deeper insight. *Decision Support Systems*, *146*, 113560. https://doi.org/10.1016/j.dss.2021.113560

Pinto, J. K. (2021). Project Management. Achieving Competitive Advantage, 5th edition, Indianapolis: Pearson.

Rye, S., Aktas, E. (2022). A Multi-Attribute Decision Support System for Allocation of Humanitarian Cluster Resources Based on Decision Makers' Perspective. *Sustainability*, 14(20), 13423. https://doi.org/10.3390/su142013423

Sabbaghtorkan, M., Batta, R., & He, Q. (2020). Prepositioning of assets and supplies in disaster operations management: Review and research gap identification. *European Journal of Operational Research*, 284(1), 1-19. https://doi.org/10.1016/j.ejor.2019.06.029

Song, B. D., Jun, S., & Lee, S. (2024). Integrated System Design for Post-Disaster Management: Multi-Facility, Multi-Period, and Bi-Objective Optimization Approach. *Systems*, *12*(3), 69. https://doi.org/10.3390/systems12030069

Wang, Y. (2021). Multiperiod Optimal Allocation of Emergency Resources in Support of Cross-Regional Disaster Sustainable Rescue. *International Journal of Disaster Risk Science*, 12(3), 394-409. https://doi.org/10.1007/s13753-021-00347-5