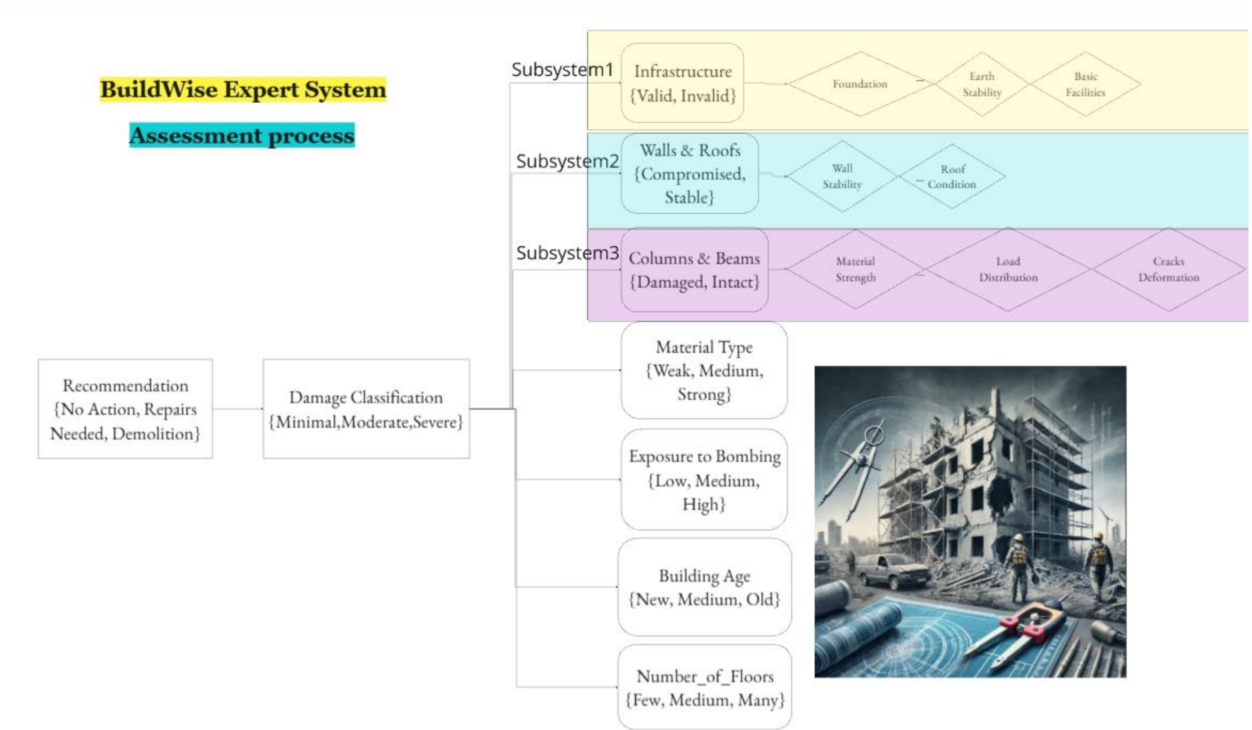
BuildWise Expert System

Introduction

Due to the massive destruction inflicted upon the Gaza Strip by the war, there is an urgent need for smart and innovative methods to aid in reconstruction and post-conflict recovery. One such solution is the BuildWise Expert system, an expert system designed to assist engineers in accelerating and enhancing the process of assessing damaged buildings. The aftermath of war often leaves infrastructure uncertain, which this system addresses through fuzzy logic. The system facilitates efficient and effective recovery efforts by providing accurate damage assessments and tailored recommendations.

Methodology (Fuzzy Reasoning)

This fuzzy expert system for evaluating building conditions follows a systematic approach broken down into the following steps [This Link of Process](#):



1. **Definition of Linguistic Variables and Membership Functions**

- Purpose: To model a fuzzy logic system's input and output parameters in linguistic terms.
- Process:
  - Define input variables such as Foundation Condition, Earth Stability, Basic Facilities, Number of Floors, Building Age, etc.
  - Define output variables like Infrastructure Validity, Walls & roof status, Columns & Beams Integrity, Damage Level, and Recommendations.

  - Use membership functions (e.g., triangular, and trapezoidal) to fuzzify the crisp inputs into linguistic categories like "Poor," "Moderate," "Good," etc.

2. Rule Creation

- **Purpose:** Establish relationships between inputs and outputs using fuzzy logic rules.
- **Process:**
  - Define rules for each subsystem, such as *Infrastructure*, *Walls and Roofs*, and *Columns and Beams*. For instance:
    - If the *Foundation* is "Poor" AND *Earth Stability* is "Unstable," THEN *Infrastructure* is "Invalid."
    - If *Wall Stability* is "Weak" OR *Roof Condition* is "Poor," THEN *Walls_Roofs* are "Compromised."
  - rules to handle variables like *Number of Floors*, *Building Age*, *Material Type*, and *Exposure to Bombing* to refine the assessment.
  - Combine subsystem outputs to evaluate the overall *Damage Level* and generate *Recommendations*.

3. Fuzzy Control System Construction

- **Purpose:** Create a functional control system to simulate decision-making based on rules and inputs.
- **Process:**
  - Build the control system by compiling the defined rules.
  - Initialize the system using skfuzzy's ControlSystem and ControlSystemSimulation modules.

4. Input Data Simulation

- **Purpose:** Simulate real-world scenarios to evaluate the system.
- **Process:**
  - Accept inputs for all defined variables (e.g., *Foundation Condition = 0.8*, *Number of Floors = 5*).

- Use the fuzzy control system to compute the results based on the defined rules.

5. Output Classification

- **Purpose:** Translate fuzzy outputs into meaningful classifications.
- **Process:**
  - Classify the *Damage Level* based on percentage thresholds:
    - Damage ≥ 70% → *Total Damage*.
    - 30% ≤ Damage < 70% → *Partial Damage*.
    - Damage < 30% → *No Damage*.
  - Generate corresponding *Recommendations* such as "Demolition," "Repair Needed," or "No Action."

Results (Forward Chaining):

Case1:

Inputs:

The building has excellent foundations (0.9), stable earth (0.8), and fully operational basic facilities (0.9). It is a 3-floor building, only 5 years old, made of strong materials (0.9), with very low exposure to bombing (0.1). Walls and roofs are in excellent condition (0.9), and structural elements like material strength (0.9) and load distribution (0.9) are strong, with minimal cracks (0.1).

Outputs:

The system predicts 17.50% damage, which is classified as No Damage. The recommendation is "No Recommendation," with a low cost for minor maintenance.

The building is in excellent condition due to its strong structural elements, new age, and minimal exposure to external risks. Only minor maintenance is needed to keep it in top shape.

Case2:

Inputs:

The building has strong foundations (0.9), stable earth (0.8), and operational basic facilities (0.7). It is a 3-floor building, 10 years old, made of strong materials (0.9), with low exposure to bombing (0.2). Walls and roofs are stable (0.85 and 0.8), and structural elements like material strength (0.75) and load distribution (0.8) are good, with minor cracks (0.3).

Outputs:

The system predicts 30.02% damage, which is classified as Partial Damage. The recommendation is "Need Repair," and partial repairs are recommended at a medium cost.

Despite the building's overall good condition, minor issues (e.g., cracks, and age) contribute to partial damage, which requires repairs to maintain safety and functionality.

Case3:

Inputs:

The building has moderate foundations (0.5), somewhat stable earth (0.6), and basic facilities that need repair (0.4). It is a seven-story, 50-year-old building made of medium-strength materials (0.6), and it has been exposed to bombing (0.7). Walls and roofs are moderately stable (0.5 and 0.6), but structural elements like material strength (0.5) and load distribution (0.6) are weak, and there are significant cracks (0.5).

Outputs:

The system predicts 77.98% damage, classified as Total Damage. The recommendation is "Demolition" with a high cost for full reconstruction.

The building's age, exposure to bombing, and structural weaknesses (e.g., cracks, and moderate material strength) have caused severe damage, making it unsafe and requiring complete demolition and reconstruction.

Results (Backward Chaining):

The same Input for these 3 Cases but with different Goals:

- Input (Known Facts):

  Foundation: Poor, Earth Stability: Unstable, Basic Facilities: Not Operational

  Walls and Roofs: Compromised, Columns and Beams: Damaged

- Case 1 Goal (Damage Level Severe):

- Output:

  Goal 'Damage_Level_Severe' is proven.

  Path: Columns_Beams_Damaged -> Walls_Roofs_Compromised -> Basic_Facilities_Not_Operational -> Earth_Stability_Unstable -> Foundation_Poor -> Infrastructure_Invalid -> Damage_Level_Severe

- Interpretation:

  The system determined that the building had severe damage based on the following reasoning:

  - The columns and beams are damaged, and the walls and roofs are compromised, indicating structural instability.
  - The basic facilities are not operational, and the earth's stability is unstable, weakening the building's infrastructure.
  - The foundation is poor, which is a critical factor in determining the overall infrastructure's validity.

- Case 2 Goal (Damage Level Minimal):

- Output: Goal 'Damage_Level_Minimal' cannot be proven.

- Interpretation:

  The system could not prove that the building has minimal damage because:

  - The known facts indicate significant issues, such as a poor foundation, unstable earth, non-operational facilities, compromised walls and roofs, and damaged columns and beams.
  - These factors collectively suggest serious structural problems, making it impossible to conclude that the damage level is minimal.

Case 3 Goal (Recommendation_Demolition):

- Outputs:

  Goal 'Recommendation_Demolition' is proven.

  Path: Columns_Beams_Damaged -> Walls_Roofs_Compromised -> Basic_Facilities_Not_Operational -> Earth_Stability_Unstable -> Foundation_Poor -> Infrastructure_Invalid -> Damage_Level_Severe -> Recommendation_Demolition

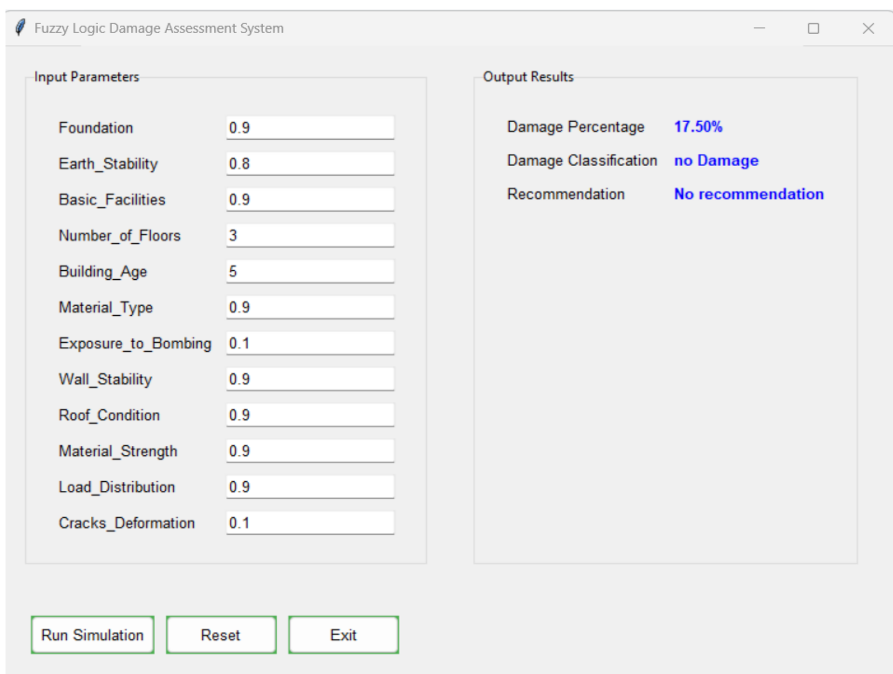
- Interpretation:
  - The system determined that demolition is recommended based on the following reasoning Combining these Inputs, the system concluded that the infrastructure is invalid, leading to a severe damage level.
  - Finally, the system recommended demolition because the damage is too severe to repair safely or cost-effectively.

Comment

The BuildWise system effectively assesses building damage and provides recommendations using fuzzy logic However, there are opportunities for future improvements to enhance its functionality and usability:

- Enhanced Rules: The system could benefit from more detailed rules This would improve its accuracy in complex scenarios.
- User Interface: A simple GUI would make the system more accessible to non-technical users.

  I made a simple one:



- Deployment: Future work could involve deploying the system as a web or mobile application, making it accessible for real-time damage assessment in the field.

Conclusion

The BuildWise system provides a solid foundation for automating building damage assessment and recommendation generation.

the system can be further enhanced to become a powerful tool for engineers, architects, and disaster response teams. Future work, such as developing a GUI and expanding the knowledge base, will make the system more user-friendly and capable of handling complex real-world scenarios.

References

- فن تقييم أضرار المباني
- دليل تقييم المباني
- خطوات فحص الابنية بعد الحرب