An evaluation of methods for prioritizing software requirements (1997)

Tom Rochette <tom.rochette@coreteks.org>

December 21, 2025 — 77e1b28a

0.1 Context

0.2 Learned in this study

0.3 Things to explore

- Minimal spanning tree to AHP hybrid approach
- Minimal spanning tree to Monte Carlo comparisons (random comparisons) approach

1 Overview

- Comparing less requirements can be traded for redundancy/robustness of the prioritization results. In other words: speed or robustness, pick one.

2 1. Introduction

- 6 candidates
    1. Analytic hierarchy process (AHP)
    2. Hierarchy AHP
    3. Spanning tree matrix
    4. Bubble sort
    5. Binary search tree
    6. Priority groups
- AHP appears to be the most promising for small as well as large scale use.

3 2 Motivation

- In AHP, decision makers pair-wise compare the requirements to determine which of the two is more important, and to what extent.
- This approach has been experienced as being effective, accurate and also to yield informative and trustworthy results.
- The major drawback of AHP is that all unique pairs of requirements are to be compared, thus the required effort can be substantial.

4 3 Prioritizing methods

- A prioritizing session may consist of three consecutive stages:

1. The preparation stage where a person structures the requirements according to the principle of the prioritizing methods to be used. A team and a team leader for the session is selected and provided all necessary information.
2. The execution stage where the decision makers do the actual prioritizing of the requirements using the information they were provided with in the previous stage. The evaluation criteria must be agreed upon by the team before the execution stage is initiated.
3. The presentation stage where the results of the execution are presented for those involved. Some prioritizing methods involve different kinds of calculations that must be carried out before the results can be presented.

4.1 3.1 The analytic hierarchy process (AHP)

- Requires $\frac{n \cdot (n-1)}{2}$ pair-wise comparisons.
- AHP is a demanding method due to the dramatically increasing number of required pair-wise comparisons when the number of requirements grows.
- AHP is very trustworthy since the huge amount of redundancy in the pair-wise comparisons makes the process fairly insensitive to judgmental errors.
- The resulting priorities are relative and based on a ratio scale, which allows for useful assessments of requirements.
- An AHP prioritizing session
    1. As preparation, outline all unique pairs of requirements
    2. As execution, compare all outlined pairs of requirements using the scale in Table 1.
    3. As presentation, use the "averaging over normalized columns" method to estimate the relative priority of each requirements. Calculate the consistency ratio of the pair-wise comparisons using methods provided by AHP. The consistency ratio is an indicator of the reliability of the resulting priorities, and thus also an estimate of the judgmental errors in the pair-wise comparisons.

Table 1: Fundamental scale used for pair-wise comparisons

Intensity of importance	Description
1	Of equal importance
3	Moderate difference in importance
5	Essential difference in importance
7	Major difference in importance
9	Extreme different in importance
Reciprocals	If requirement i has one of the above numbers assigned to it when compared with requirement j, then j has the reciprocal value when compared with i

4.2 3.2 Hierarchy AHP

- Instead of comparing requirements as if they were all of the same type, compare requirements with similar abstraction. The more general requirements are at the top while the more specific requirements are below them. Priorities are propagated from top to bottom.

4.3 3.3 Minimal spanning tree

- If decision makers were perfectly consistent, the redundancy of the comparisons would be unnecessary.
- In such case, only $n-1$ comparisons would be enough to calculate the relative intensity of the remaining comparisons.
- This implies that the least effort required by a decision maker is to create a minimal spanning tree in a directed graph.

- Minimal spanning tree prioritization session
    1. As preparation, outline $n - 1$ unique pairs of requirements so that a minimal spanning tree can be constructed.
    2. As execution, compare all outlined pairs of requirements using the scale in Table 1
    3. As presentation, compute the missing intensities of importance by taking the geometric mean of the existing intensities of all possible ways in which they are connected. Then use AHP as usual.

4.4 3.4 Bubble sort

- Similar to AHP, $\frac{n \cdot (n-1)}{2}$ pair-wise comparisons, but no comparison about the extent of the difference.

4.5 3.5 Binary search tree

- Requires $O(n \log n)$ pair-wise comparisons.
- Build a binary search tree by comparing the requirement with the requirements already listed in the table.

4.6 3.6 Priority groups

- Compare requirements within priority groups
- Priority groups prioritization session
    1. As preparation, outline the candidate requirements.
    2. As execution, put each of the requirements into one of the three groups (low, medium, high). In groups with more than one requirements, create three new subgroups and put the requirements into these groups. Continue to apply this process recursively to all groups.
    3. As presentation, just read the requirements from left to right.
- The head-tail (highest ranked and lowest ranked) of each groups can be compared to make sure that the requirements are in the correct order.

4.7 4.3.1 Inherent characteristics

- Two inherent characteristics of the prioritizing methods were identified:
    - Consistency indication: this characteristic indicates whether the prioritizing method is able to indicate consitency in the decision maker's judgment. This ability requires redundancy in the judgements.
    - Scale of measurement: this characteristic describes the scale on which the resulting requirements priorities are based. The scale used for ranking the requirements is an important attribute of goodness. The more powerful the scale, the more useful the assessment of the requirements can be carried out. These measurement scales in increasing order of strength are: nominal, ordinal, interval, and ratio scales.

4.8 4.3.2 Objective measures

- Required number of decisions
- Total time consumption
- Time consumption per decision

4.8.1 4.3.3 Subjective measures

- Ease of use
- Reliability of results
- Fault tolerance

4.9 4.5 Threats to validity

- Requirements are interdependent

4.10 6. Discussion and recommendation

- The ratio scale used by AHP provides priority distance between requirements which becomes very tangible, the other methods only providing the correct order.

5 See also

6 References

- Karlsson, Joachim, Claes Wohlin, and Björn Regnell. "An evaluation of methods for prioritizing software requirements." Information and Software Technology 39.14 (1998): 939-947.