VEHICLE ELECTRONICS AND ARCHITECTURE

SOFTWARE RELIABILITY PREDICTION FOR ARMY VEHICLE

Macam S Dattathreya
US Army RDECOM-TARDEC, Warren, MI

Harpreet Singh
Wayne State University, Detroit, MI

AGENDA

- Objective
- Approach
- AVS reliability metrics
- Prediction algorithm
- Summary

OBJECTIVE

- Formulate Army vehicle software (AVS) reliability metrics
- Develop AVS reliability prediction technique

APPROACH

- Formulate AVS reliability metrics
  - Investigate IT architecture documents
  - Capture details.
    - Data characteristics (e.g., format, size, storage, and encryption)
    - Inputs and outputs
    - Test cases
    - Configuration and Fault handling
  - Formulate metrics
    - Quantify
- Develop AVS reliability prediction technique
  - Fuzzy logic
  - Fuzzy sets

IT ARCHITECTURE DOCUMENTS

- Transform user requirements into implementation
- Pure text or Unified Modeling Language (UML)
- No implementation details
- Guide for designers and developers

FUZZY LOGIC

- Approximation technique for imprecise situations
  - Handle vagueness using heuristic technique (expert knowledge)
  - Fuzzy set theory based (Lotfi Zadeh)
  - Linguistic terms usage
    - Hot, cold, very tall, high reliability
  - Expert knowledge rules in linguistic terms
    - If more defects reliability is low
      - Linguistic terms = more and low
- Fuzzy sets
  - Elements with different membership grades between 0 and 1
  - If X is a set denoted by Y, then a fuzzy set S in X is a set of ordered pairs
    - S = {(x, μY(x)) | x ε X} where μ is a membership function
    - Example: S = {(7', 0.9), (7'5", 1), (6'5", 0.8), (6', 0.7), (5', 0.3)}

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
22 JUN 2100	**N/A**	**-**

4. TITLE AND SUBTITLE
Software Reliability Prediction for Army Vehicle

5a. CONTRACT NUMBER
5b. GRANT NUMBER
5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
Macam S. Dattathreya; Harpreet Singh

5d. PROJECT NUMBER
5e. TASK NUMBER
5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
US Army RDECOM-TARDEC 6501 E 11 Mile Rd Warren, MI 48397-5000, USA

8. PERFORMING ORGANIZATION REPORT NUMBER
21947

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
US Army RDECOM-TARDEC 6501 E 11 Mile Rd Warren, MI 48397-5000, USA Wayne State University, Detroit, MI, USA

10. SPONSOR/MONITOR'S ACRONYM(S)
TACOM/TARDEC/RDECOM

11. SPONSOR/MONITOR'S REPORT NUMBER(S)
21947

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release, distribution unlimited

13. SUPPLEMENTARY NOTES
Presented at the 2011 NDIA Vehicles Systems Engineering and Technology Symposium 9-11 August 2011, Dearborn, Michigan, USA, The original document contains color images.

14. ABSTRACT

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			
unclassified	**unclassified**	**unclassified**	**SAR**	**3**	

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std Z39-18

AVS RELIABILITY METRICS

- Data handling (D)
  - Data and its characteristics
  - Test cases
- Interoperability (I)
  - Exchange data within predefined access restrictions
  - Inputs & outputs
  - Test cases
- Configurability (C)
  - Multiple operating environments
  - Test cases
- Fault handling (F)
  - Fault handling mechanisms
  - Test cases

DATA HANDLING (D)

- D_1 = # of distinct data elements
- D_2 = # of distinct data elements captured with necessary details
- D_3 = # of distinct data elements that have captured required data characteristics
- T_1 = total test cases for all the data elements
- T_{1I} = # of test cases that are planned for testing all data characteristics per data element
- N_c = total # of data characteristics

$$T1 = \sum_{I=1}^{D_1} \frac{\sum_{i=1}^{N_c} T_{1Ii}}{N_c}$$

$$D = 3 - \left(\frac{D_2 + D_3}{D_1} + \frac{T_1}{D_1} \right)$$

INTEROPERABILITY (I)

- I_1 = # required distinct inputs O_1 = # of required distinct outputs
- I_2 = # of distinct inputs captured with details
- O_2 = # of distinct outputs captured with details
- T_2 = total test cases for all the input details
- T_{2I} = # of test cases that are planned for testing all inputs details/input
- T_{3I} = # of test cases that are planned for testing all output details/output
- N_i = total # of input details, N_o = total # of output details
- T4 = # distinct inputs planned for testing event logging
- T6 = # distinct outputs planned for testing event logging
- T8 = # distinct inputs planned for testing fault handling
- T10 = # distinct outputs planned for testing fault handling

$$T_2 = \sum_{I=1}^{I_1} \frac{\sum_{i=1}^{N_i} T_{2Ii}}{N_i} \qquad I = 8 - \left(\frac{I_2}{I_1} + \frac{T_2 + T_4 + T_8}{I_1} + \frac{O_2}{O_1} + \frac{T_3 + T_6 + T_{10}}{O_1} \right) \qquad T_3 = \sum_{I=1}^{O_1} \frac{\sum_{i=1}^{N_o} T_{3Ii}}{N_o}$$

CONFIGURABILITY (C)

- C_1 and C_2 are the number of distinct inputs and outputs, respectively planned for configurable event logging
- C_3 and C_4 are the number of distinct inputs and outputs, respectively planned for configurable fault handling
- T_5 and T_7 are the number of distinct inputs and outputs, respectively planned for testing its configurable event logging
- T_9 and T_{11} are the number of distinct inputs and outputs, respectively planned for testing its configurable fault handling
- I_1 = # required distinct inputs O_1 = # of required distinct outputs
- I_2 = # of distinct inputs captured with details
- O_2 = # of distinct outputs captured with details

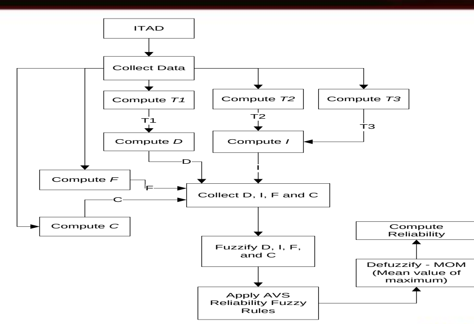
$$C = 8 - \left(\frac{C_1 + C_3}{I_1} + \frac{T_5 + T_9}{I_1} + \frac{C_2 + C_4}{O_1} + \frac{T_7 + T_{11}}{O_1} \right)$$

FAULT HANDLING (F)

- E_1 = number of distinct inputs planned for event logging
- F_1 = number of distinct inputs planned for fault handling
- E_2 = number of distinct outputs planned for event logging
- F_2 = number of distinct outputs planned for fault handling
- T_8 = number of distinct inputs planned for testing its fault handling
- T_{10} = number of distinct outputs planned for testing its fault handling
- I_1 = # required distinct inputs
- O_1 = # of required distinct outputs

$$F = 6 - \left(\frac{E_1 + F_1}{I_1} + \frac{E_2 + F_2}{O_1} + \frac{T_8}{I_1} + \frac{T_{10}}{O_1} \right)$$

PREDICTION ALGORITHM

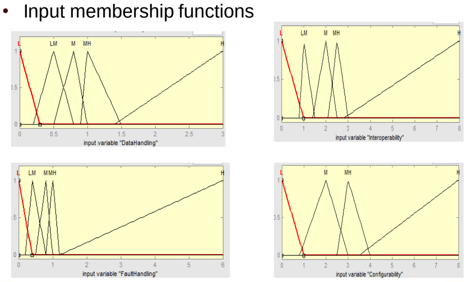


ALGORITHM: Main Steps

- Fuzzify (fuzzification) inputs
- Apply expert knowledge based rules
- Defuzzify (defuzzification)
- Predict AVS reliability

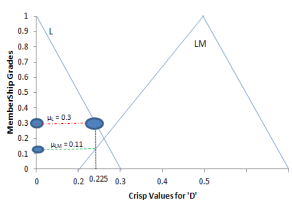
FUZZIFICATION

- Map crisp inputs to membership grades
- Input membership functions



FUZZIFICATION: Continued

- 'D' = 0.225
- max (μ_{LM}, μ_L) = max(0.3, 0.11) = 0.3



APPLY RULES

- Fuzzy reasoning - aggregation of results
- Maximum of mean value
- 'and' operator D ∧ I ∧ F ∧ C = min (μ_D μ_I μ_F μ_C)
- 'or' operator D ∧ I ∧ F ∧ C = max (μ_D μ_I μ_F μ_C)

1. If (D is L) and (I is L) and (F is L) and (C is L) then (Reliability is H) (1)
2. If (D is L) and (I is LM) and (F is L) and (C is L) then (Reliability is H) (1)
3. If (D is L) and (I is L) and (F is LM) and (C is L) then (Reliability is H) (1)
4. If (D is L) and (I is L) and (F is L) and (C is M) then (Reliability is H) (1)
5. If (D is LM) and (I is L) and (F is L) and (C is L) then (Reliability is H) (1)
6. If (D is LM) and (I is LM) and (F is L) and (C is L) then (Reliability is M) (1)
7. If (D is LM) and (I is L) and (F is LM) and (C is L) then (Reliability is M) (1)
8. If (D is LM) and (I is LM) and (F is L) and (C is L) then (Reliability is M) (1)
9. If (D is LM) and (I is L) and (F is L) and (C is M) then (Reliability is M) (1)
10. If (D is LM) and (I is L) and (F is LM) and (C is M) then (Reliability is M) (1)
11. If (D is M) and (I is L) and (F is L) and (C is L) then (Reliability is M) (1)
12. If (D is MH) or (I is H) or (F is H) or (C is MH) then (Reliability is L) (1)
13. If (D is H) or (I is MH) or (F is MH) or (C is H) then (Reliability is L) (1)

FUZZY RULES AND DEFUZZIFICATION

SUMMARY

- Concept introduction
  - AVS reliability metrics
    - IT Architecture documents
  - AVS reliability prediction algorithm
    - Approximation
    - Fuzzy logic
- Simple data collection
- Ordinary computer skill