Anggraini, R. N. E., & Martin, T. P. (2019). Capturing Requirement Correlation in Adaptive Systems. In *2018 International Conference on Robotics, Biomimetics, and Intelligent Computational Systems, (Robionetics 2018)* (pp. 6-11). [8674673] Institute of Electrical and Electronics Engineers (IEEE). https://doi.org/10.1109/ROBIONETICS.2018.8674673

Capturing Requirement Correlation in Adaptive Systems

Ratih N.E. Anggraini
Intelligent Systems Lab, University of Bristol
Bristol, United Kingdom
ra16032@bristol.ac.uk

T.P. Martin
Intelligent Systems Lab, University of Bristol
Bristol, United Kingdom
trevor.martin@bristol.ac.uk

***Abstract*—An adaptive system is expected to modify its behavior to suit changes in environmental and system condition. Somehow, it causes problems during requirement specification (and in subsequent verification) since it is difficult to provide all possible adaptation needed during runtime. Thus, it may be necessary to temporarily ignore non-critical requirements to a certain point in order to maintain satisfaction, especially on critical and invariant requirements.**

One way to handle uncertainty in the adaptive system is by relaxing requirement and present the verification result using a graded (fuzzy) condition in a requirement satisfaction. Often, relaxing one requirement can affect the satisfaction of another related requirement. In this paper, we use linear regression to capture the relationship between two relaxed system requirements. Pearson and Spearman correlation coefficient is utilized to calculate correlation strength. To illustrate the approach, we consider a smart vacuum system problem.

Keywords—adaptive system; relax requirement, linear regression; Pearson correlation coefficient; Spearman correlation coefficient

I. INTRODUCTION

Adaptive systems are supposed to keep satisfying its requirements by doing adaptation in response to a changing environmental and system conditions [1]. The more complex the environment, the more uncertainties arise. Thus, uncertainty on adaptive systems has become major concerns on many works [2-5].

To deal with uncertainty, especially in requirement specification phase, Whittle et.al proposed a requirement language called RELAX [6]. RELAX incorporated several types of operators to address uncertainty in system properties. The verification result on the system, then can be described as fuzzy satisfaction to illustrate its degree of satisfaction [7].

Relaxing a requirement satisfaction on adaptive system can become an answer in handling uncertainty. Somehow it also can affect related requirements, either positively or negatively. Souza et. al introduced Awareness Requirements, requirements that talk about other requirements success or failure [8]. It emphasized the importance on monitoring requirements at runtime to provide feedback loops.

Correlation describes relationship, involving dependency, between two variables and commonly examine how close is the linear relationship with each other. Correlation is useful to predict future association between those variables. Linear regression, Pearson and Spearman correlation coefficient are simplest way in capturing the relationship between two variables. It was widely used in medical and psychological researches [9-12].

In this paper we adopt linear regression to capture correlation between two related requirements in adaptive system. Pearson and Spearman correlation coefficient is used to describe the relationship strength. A case study was used to demonstrate the approach.

II. BACKGROUND

A. Requirement Relaxation in Adaptive System

The ability of adaptive system to adapt to changing environment led to the growth of uncertainties. Preparing all explicit states during system lifetime become impossible. Thus, tolerate environment conditions, especially a non-critical one, to a certain point is necessary.

A requirement language called RELAX provide a structured natural language to capture uncertainty in adaptive system [13]. The requirement is written using RELAX grammar shown in (1). It incorporated modal, temporal and ordinal operators with uncertainty factors. RELAX operators are shown in Fig 1. The phrase 'AS POSSIBLE' in temporal and ordinal operator facilitate uncertainty which mean the requirement satisfaction can be tolerate to a defined threshold.

$$\begin{aligned}\phi ::= &\, true \mid false \mid P \mid SHALL\ \phi \mid \text{MAY } \phi_1 \text{ OR MAY } \phi_2 \\ &\mid EVENTUALLY\ \phi \mid \phi_1 \text{UNTIL } \phi_2 \mid BEFORE\ e\ \phi \mid \\ &\text{AFTER e } \phi \mid \text{IN t } \phi \mid \text{AS CLOSE AS POSSIBLE f } \phi \mid \\ &\text{AS \{EARLY. LATE. MANY. FEW\} AS POSSIBLE } \phi \qquad (1)\end{aligned}$$

Table I shows the requirements on Adaptive Assisted Living system written in RELAX language. Requirement R1 is divided into six subset requirements. R1.1 to R1.4 are relax requirements while R1.5 and R1.6 are invariant one.

The DEP factor indicates the impact of relaxing one requirement to the others. For example, relaxing R1.1 will disturb R1.2 to adjust diet plan but will help minimizing battery consumption and latency. Somehow the description is unclear, so we hardly know how big the effect of relaxing a requirement to the related one.

To verify requirement of adaptive system written in RELAX language, Anggraini et.al proposed fuzzy satisfaction [7]. It used UPPAAL model checker [14] to model and simulate the problem. The simulation result from UPPAAL was translated into graded satisfaction using fuzzy function.

Somehow relaxing a requirement does not as simple as it is. There is an issue when we relax a requirement that relate to another requirement since it can affect related requirement, either positively or negatively. Thus, it is important to understand the relationship on the requirements of adaptive system before we decide how much we need relaxation.

B. Linear Regression and Correlation

Linear regression analysis estimates the relationship of one variable to another by illustrating it in the slope of regression line [15]. Linear regression function of independent variable x_i, for $i = 1, \dots, n$ is shown in (2). The intercept α is where the regression line cross *y-axis* and β is the slope showing how steep the line is and ε_i is random error component.

$$y_i = \alpha + \beta x_i + \varepsilon_i \tag{2}$$

Correlation coefficient is a means to describe relationship strength between two continuous variables [16, 17]. It has value between -1 to +1. Negative correlation means that every increase in one variable will decrease another, 0 means both variables is not correlated, and positive value means that every increase in one variable will increase another.

Pearson correlation coefficient (r) is employed to measure linear relationship between two random variables [18]. It can be computed on a sample data. Suppose we have n dataset with the first sample dataset ($x_1, \dots, x_n$) and the second sample dataset ($y_1, \dots, y_n$), the Pearson correlation coefficient r can be calculated using (3).

$$r = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}} \tag{3}$$

In formula (3), $\bar{x}$ and $\bar{y}$ are the mean values of x_i and y_i, respectively and can be calculated using (4).

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i \tag{4}$$

Cohen's effect size is widely used to interpret the result of Pearson correlation r [19]. The guidelines of using effect size to evaluate relationship strength is described in Table II. The small size effect can be implied that it really is happening but will need a very careful study to see it through. Whereas a large one means that you can see the impact 'just with a naked eyes'.

TABLE I. AAL REQUIREMENTS [13]

R1: The system SHALL monitor Mary's health and SHALL notify emergency services in case of emergency.	
R1.1	The fridge SHALL The fridge SHALL detect and communicate information with AS MANY food packages AS POSSIBLE. ENV : Food locations, food item information (type, calories) & food state (spoiled, unspoiled). MON : RFID readers; Cameras; Weight sensors. REL : RFID tags provide food locations/food information/food state; Cameras provide food locations; Weight sensors provide food information (whether eaten or not). DEP : R1.1' negatively impacts R1.2'; R1.1' positively impacts R1.4 and R1.6
R1.2	The fridge SHALL suggest a diet plan with total calories AS CLOSE AS POSSIBLE TO the daily ideal calories. The fridge SHALL adjust the diet plan in line with Mary's actual calorie consumption. ENV : Mary's daily calorie consumption. MON : RFID readers and weight sensors in fridge and trash can. REL : RFID readers and weight sensors provide consumed items; items vanish from fridge and the items (if uneaten) or the packaging (if eaten) appears in trash can. DEP : R1.2' is negatively impacted by R1.1'; R1.2'negatively impacts R1
R1.3	The system SHALL ensure that Mary's liquid intake is AS CLOSE AS POSSIBLE TO the required minimum volume during the course of the day. The system SHALL ensure minimum liquid intake BEFORE bedtime. ENV : Mary's daily liquid intake. MON : fluid monitoring cups; orientation sensor-enabled cups; faucet sensors; flowerpot moisture sensors; timers correlating temporal events of different sensors: was cup emptied down sink, into flower pot or did Mary drink from it? REL : cup sensors & moisture sensors & faucet sensors & sink outlet sensors & timers all interact to collaboratively determine Mary's daily liquid intake. DEP : R1.3 negatively impacts R1.
R1.4	The system SHALL consume AS FEW units of energy AS POSSIBLE during normal operation. ENV : Total energy consumption. MON : Smart energy monitors. REL : Smart energy monitors can sense device energy consumption and sense activity within the AAL and use these to control (e.g.) lighting and heating. DEP : R1.4' is negatively impacted by R1.6
R1.5	The system SHALL raise an alarm if no activity by Mary is detected for t.b.d. hours during normal waking hours.
R1.6	The system SHALL minimize latency when an alarm has been raised.

TABLE II. EFFECT SIZE GUIDE FOR PEARSON *R*

Pearson *r*	Effect size
0.1	Small
0.3	Medium
0.5	Large

RELAX operator	Description
Modal operators	
SHALL	A requirement must hold
MAY...OR	A requirement specifies one or more alternatives
Temporal operators	
EVENTUALLY	A requirement must hold eventually
UNTIL	A requirement must hold until a future position
BEFORE, AFTER	A requirement must hold before or after a particular event
IN	A requirement must hold during a particular time interval
AS EARLY, LATE AS POSSIBLE	A requirement specifies something that should hold as soon as possible or shouldbe delayed as long as possible
AS CLOSE AS POSSIBLE TO [frequency]	A requirement specifies something that happens repeatedly but the frequency may be relaxed
Ordinal operators	
AS CLOSE ASPOSSIBLE TO [quantity]	A requirement specifies a countable quantity but the exact count may be relaxed
AS MANY, FEWAS POSSIBLE	A requirement specifies a countable quantity but the exact count may be relaxed
Uncertainty factors	
ENV	Defines a set ofproperties that define the system's environment
MON	Defines a set of properties that can be monitored by the system
REL	Defines the relationship between the **ENV** and **MON** properties
DEP	Identifies the dependencies between the (relaxed and invariant) requirements

Fig. 1. RELAX operators [13]

Squaring the value of Pearson correlation coefficient or usually called coefficient of determination (R^2) is worthwhile to explain the proportion of variance in the variable. It provides the information on how good a predictor might be constructed from the modeled value. The value of R^2 is range from 0 to 1. An R^2 of 0 means that the dependent variable unable to be predicted from the independent variable and R^2 of 1 means that it can be predicted without error.

Sometimes sample data are not normally distributed or there are several outliers that distorted the relationship of two random variables. At this condition, using Spearman's rank correlation can show a better result on the association strength rather than Pearson's correlation coefficient. Spearman's correlation coefficient of n data ranks, denoted r_s or ρ, can be computed using (5) [20].

$$r_s = 1 - \frac{6\sum d_i{}^2}{n(n^2-1)} \quad (5)$$

The value of d_i is the difference between ranks $rg(x_i)$ and $rg(y_i)$ calculated using (6).

$$d_i = rg(x_i) - rg(y_i) \quad (6)$$

III. An Example: Smart Vacuum Systems

Smart vacuum systems (SVS) problem [21] was used to illustrate requirement correlation. The problem definition of SVS is as follows:

> *An SVS is a robot which is able of cleaning a required area and balancing path plan and battery consumption. The robot has some sensors such as bumper sensor to prevent from collision and motor sensor to inform its speed and power. The controller used data from sensors to determine optimal path and preserve battery.*
>
> *In this case, the uncertainty comes from sensor data noise and environment (the amount of dirt spread in the area and power needed to clean the area). SVS needs adaptation to provide acceptable satisfaction.*

We focused on requirements SA and SB written in RELAX requirements language as shown in Table III. Uncertainty factor DEP shows that there is dependency between SA and SB but did not explicitly described how strong is the impact. Thus, we will show how to capture the correlation strength using linear regression and correlation coefficient to know the impact of relaxing a requirement to another one.

Based on the requirements, a model (Fig 2) was built using UPPAAL model checker [22]. In this model, the vacuum will repeatedly clean up the room space with the random dirt in each area unit until the battery reach its threshold or the room is all clean. Then it will stop cleaning and calculate the percentage of clean area.

The approach to measure the requirement correlation on adaptive system is as follows:

1. Simulate the model to obtain sample data on variable x and y.
2. Plot the simulation data in scatter diagram.
3. Obtain linear regression equation using (2).

4. Analyze relationship strength using (3) and Table I.
5. Compute rank correlation using (5).

To capture the requirement relationship in SA and SB on SVS problem, a simulation has been conducted on SVS model in UPPAAL. To do so, the initial battery was set to 100 units and the initial space was 50 units, while the dirt was randomized between (1,2,3) in each space unit. The initial remaining battery threshold was set to 20 and then it was relaxed up to 0. Then the percentage of clean space was calculated to be used in the sample data.

TABLE III. SVS REQUIREMENTS

S_A	SVS SHALL achieve AS MANY clean AS POSSIBLE ENV: room space MON: motion sensor REL: motion sensor provides the room space that has been cleaned DEP: SA is negatively impacted by SB
S_B	The system SHALL have remaining battery AS MANY AS POSSIBLE ENV: remaining battery MON: number of dirt REL: number of dirt will determine how much battery power is needed DEP: SB negatively impacts SA

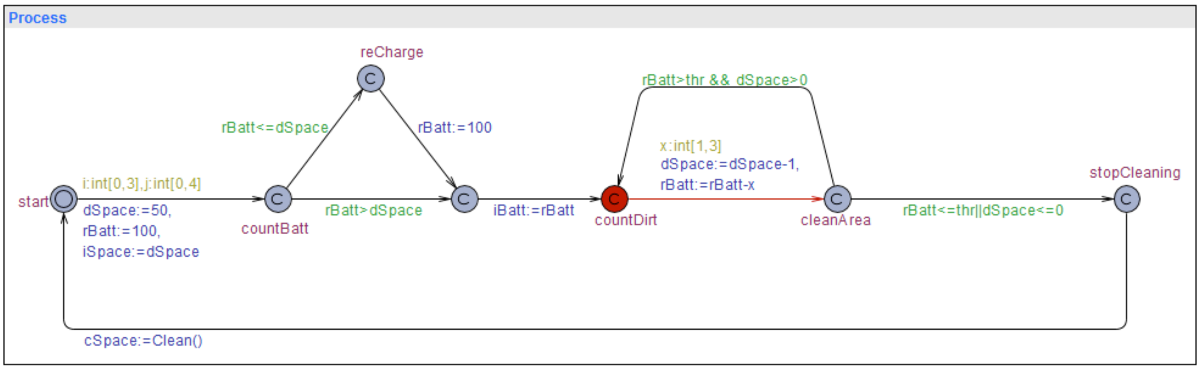

Fig. 2. SVS model on UPPAAL

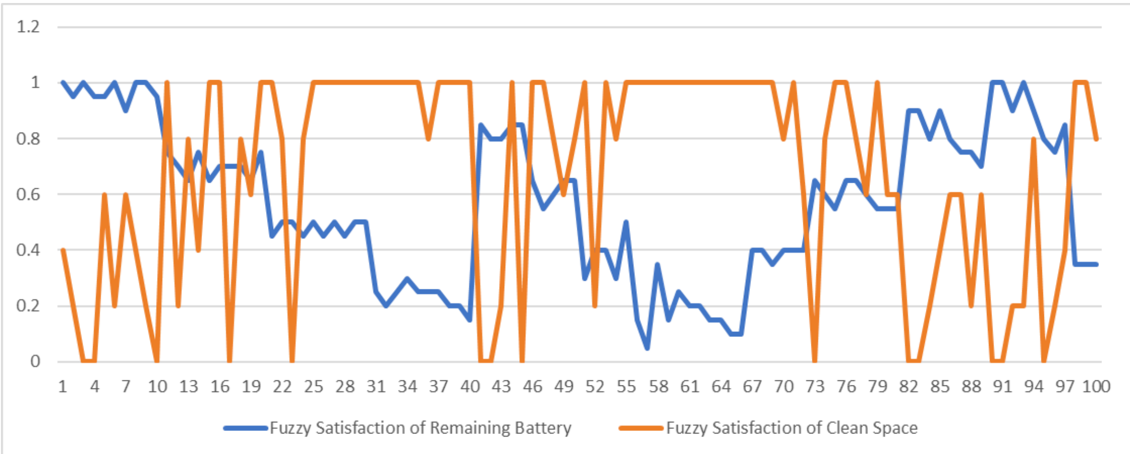

Fig. 3. Fuzzy Satisfaction of the simulation result on SVS model

Fig 3 shows the fuzzy satisfaction on simulation result[1] on UPPAAL. It addresses the negative association as mentioned in DEP factor where the value of clean space satisfaction is high when the battery threshold is relaxed more (low remaining battery value). Fig 4 is the scatter plot of simulation data (blue dot) and regression line (red line) estimated from simulation data. The trendline of linear regression is described in (7) and Pearson correlation coefficient is shown in (8).

$$y = 99.397 - 0.9203x \quad (7)$$

$$r = -0.699782848 \quad (8)$$

$$R^2 = 0.489696034 \quad (9)$$

[1] http://tiny.cc/SVS_simulation

$$r_s = -0.706885689 \quad (10)$$

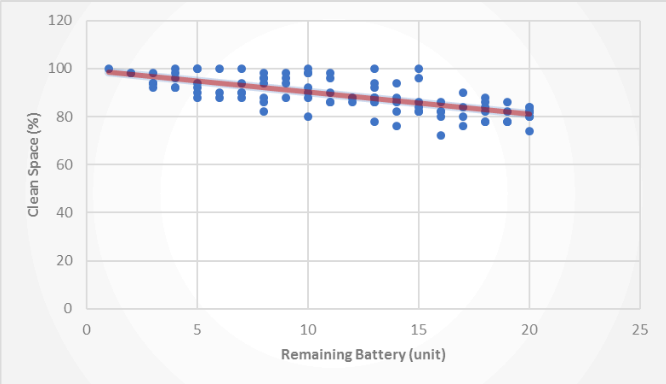

Fig. 4. SVS cleaning result with different battery relaxation

Based on linear regression function in (7), the intercept indicates that at x=0, the value of y is *99*.397 and the slope indicates a negative relationship where every unit increase in x will decrease y by average *0*.9203. Using Cohen's effect size in Table II, the value of Pearson correlation coefficient shows that SB has a large negative impact on SA. This means that the smaller the remaining battery, the more the robot can clean the space. Or in other words, we need to relax more on the remaining battery to achieve more clean space. While the R^2 shown in (9) reveal that 49% of variance in y is predictable from x.

As we can see in (8) and (10), the value of Pearson and Spearman correlation coefficient confirmed that SA and SB have a strong association. In addition, it also shows a similar result which means that the outliers do not distort the relationship on both requirements.

IV. Conclusions

An approach of capturing relationship between two requirements in adaptive system using linear regression was demonstrated herein. Pearson correlation coefficient and Spearman rank correlation coefficient were introduced to measure correlation strength. The variability of sample data was described in coefficient of determination by squaring the value of Pearson coefficient. The method was implemented on the problem of smart vacuum system using simulation data obtained from UPPAAL.

Based on current result, the direction of future works includes doing a reverse engineering on adaptive system problem to calculate relaxation needed to achieve certain requirement satisfaction. Another interesting idea is exploring if there is any causal relationship in the requirement and find the suitable method to extract this relationship. In addition, formalize the approach on requirement satisfaction and requirements correlation on dynamic adaptive system is needed.

References

1. Tamura, G., et al., *Towards Practical Runtime Verification and Validation of Self-Adaptive Software Systems*, in *Software Engineering for Self-Adaptive Systems II: International Seminar, Dagstuhl Castle, Germany, October 24-29, 2010 Revised Selected and Invited Papers*, R. de Lemos, et al., Editors. 2013, Springer Berlin Heidelberg: Berlin, Heidelberg. p. 108-132.
2. Ramirez, A.J., A.C. Jensen, and B.H.C. Cheng. *A taxonomy of uncertainty for dynamically adaptive systems*. in *2012 7th International Symposium on Software Engineering for Adaptive and Self-Managing Systems (SEAMS)*. 2012.
3. Esfahani, N., E. Kouroshfar, and S. Malek. *Taming uncertainty in self-adaptive software*. in *Proceedings of the 19th ACM SIGSOFT symposium and the 13th European conference on Foundations of software engineering*. 2011. ACM.
4. Esfahani, N. and S. Malek, *Uncertainty in Self-Adaptive Software Systems*, in *Software Engineering for Self-Adaptive Systems II: International Seminar, Dagstuhl Castle, Germany, October 24-29, 2010 Revised Selected and Invited Papers*, R. de Lemos, et al., Editors. 2013, Springer Berlin Heidelberg: Berlin, Heidelberg. p. 214-238.
5. Bencomo, N., A. Belaggoun, and V. Issarny. *Bayesian artificial intelligence for tackling uncertainty in self-adaptive systems: The case of dynamic decision networks*. in *2013 2nd International Workshop on Realizing Artificial Intelligence Synergies in Software Engineering (RAISE)*. 2013.
6. Whittle, J., et al. *RELAX: Incorporating Uncertainty into the Specification of Self-Adaptive Systems*. in *2009 17th IEEE International Requirements Engineering Conference*. 2009.
7. Anggraini, R.N. and T. Martin. *Fuzzy Representation for Flexible Requirement Satisfaction*. in *UK Workshop on Computational Intelligence*. 2017. Springer.
8. Souza, V.E.S., et al., *Awareness requirements for adaptive systems*, in *Proceedings of the 6th International Symposium on Software Engineering for Adaptive and Self-Managing Systems*. 2011, ACM: Waikiki, Honolulu, HI, USA. p. 60-69.
9. Mukaka, M.M., *A guide to appropriate use of correlation coefficient in medical research.* Malawi Medical Journal, 2012. **24**(3): p. 69-71.
10. Thirumalai, C., A. Duba, and R. Reddy. *Decision making system using machine learning and Pearson for heart attack*. in *Electronics, Communication and Aerospace Technology (ICECA), 2017 International conference of*. 2017. IEEE.

11. Gazes, R.P., R.R. Hampton, and S.F. Lourenco, *Transitive inference of social dominance by human infants.* Developmental science, 2017. **20**(2): p. e12367.
12. McLean, S.P., et al., *Applying the flanker task to political psychology: A research note.* Political Psychology, 2014. **35**(6): p. 831-840.
13. Whittle, J., et al., *RELAX: a language to address uncertainty in self-adaptive systems requirement.* Requirements Engineering, 2010. **15**(2 %@ 1432-010X): p. 177-196.
14. *UPPAAL website*. [cited 2017; Available from: http://www.uppaal.org/.
15. Montgomery, D.C., E.A. Peck, and G.G. Vining, *Introduction to linear regression analysis*. Vol. 821. 2012: John Wiley & Sons.
16. Bobko, P., *Correlation and regression: Applications for industrial organizational psychology and management*. 2001: Sage.
17. Zou, K.H., K. Tuncali, and S.G. Silverman, *Correlation and simple linear regression.* Radiology, 2003. **227**(3): p. 617-628.
18. Pearson, K., *Mathematical contributions to the theory of evolution. III. Regression, heredity, and panmixia.* Philosophical Transactions of the Royal Society of London. Series A, containing papers of a mathematical or physical character, 1896. **187**: p. 253-318.
19. Cohen, J., *A power primer.* Psychological bulletin, 1992. **112**(1): p. 155.
20. Corder, G.W. and D.I. Foreman, *Nonparametric statistics: A step-by-step approach*. 2014: John Wiley & Sons.
21. Fredericks, E.M., B. DeVries, and B.H. Cheng, *AutoRELAX: automatically RELAXing a goal model to address uncertainty.* Empirical Software Engineering, 2014. **19**(5): p. 1466-1501.
22. David, A., et al., *Uppaal SMC tutorial.* International Journal on Software Tools for Technology Transfer, 2015. **17**(4): p. 397-415.