Title	How reliable is smartness? And how smart is reliability?
Author(s)	Lee, WK
Citation	The 2013 IET International Conference on Smart and Sustainable City (ICSSC 2013), Shanghai, China, 19-20 August 2013. In the IET Conference Publication Series, v. 2013 n. 635 CP, article no. 1964, p. 381-384
Issued Date	2013
URL	http://hdl.handle.net/10722/204108
Rights	Creative Commons: Attribution 3.0 Hong Kong License

HOW RELIABLE IS SMARTNESS? AND HOW SMART IS RELIABILITY?

W. K. Lee[1]

[1]*Department of Electrical and Electronic Engineering, The University of Hong Kong, Hong Kong SAR, China*
wklee@eee.hku.hk

Keywords: Reliability, Redundancy, Cold Standby, Smart Grid, Smart City.

Abstract

This paper highlights major reliability concerns in the trend of building smartness in everything from devices to systems. It alerts engineers to determine the trade-off equilibrium of new smartness in a more practical and realistic manner. The discussion is based on several common roles of smart practices that include software; driver; and redundancy. The major concerns are expressed in five areas: series reliability shrinkage; cold standby's intrinsic imperfection; crossroad & roundabout jeopardy; software unreliability and cyber vulnerability.

1 Introduction

Nowadays smartness is a hot topic in technology. It is stirring up a new era in human history. We have everything being smart nowadays. Smart communication, smart home, smart building, smart modular technology, smart e-bike, smart grid, smart load …. , and last but not least, smart city. Engineers believe that the equipment, plant or system is smarter because there is more built-in or networking intelligence to look after operating parts and parameters. Smartness improves performance, but improved performance may not imply improved reliability. An intelligent man is an analogy. He often performs better than an average person; yet his performance does not upgrade his longevity. Very often, the enrichment of life commitments makes him more vulnerable to risks and flops. The chance of his life failure rate cannot be improved by his raw intelligence, unless such intelligence also makes him lower the "wear and tear" and "stress intolerance" which effectively lengthen the MTTF. Despite the concept of self-healing is now a part of smartness input to system, seldom will users challenge whether sacrifice of other reliabilities has been worthwhile.

2 Smart Role Model

Figure 1 depicts the five roles of smart technology in a plant performance, with a further understanding that in the modern world, this plant performance is not stand alone, but links to other plants or systems through internet connections. In this paper, plant is used as a generic term that covers load, process and system.

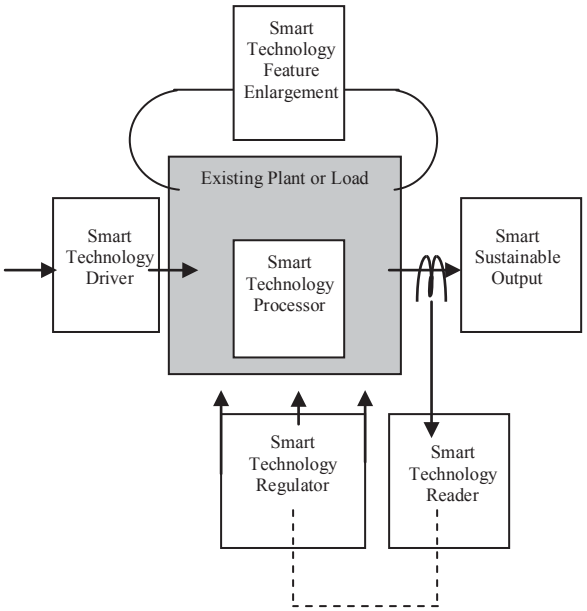

Figure 1. Smart Role Model

3 Series Reliability Shrinkage

A driver is a series component in any process. It is the first serial step in an operation. Smart driver for a plant is no exception. In order to be reliable, both the driver and the "as was" plant process must be reliable. The mathematical expression have been well developed by

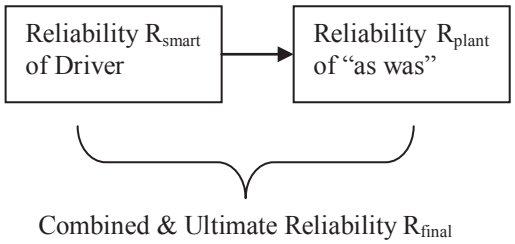

Figure 2. Series Reliability

With 2 components only in series, the combined reliability is given by:

$$R_{final} = R_{smart} \times R_{plant} \quad (1)$$

Hence a smart driver always lowers reliability unless it is a replacement for a less reliable driver. Even when it is an additional driver for smooth start and capability enhancement, still it will reduce reliability accordingly regardless the quality of performance it is elevating. The application of power

electronics drives in the late century and computer drives are examples in smart technology that enhanced capability, yet the effect on reliability was often overlooked.
Similarly once smart technology is introduced to process a plant in series operational steps, similar reliability shrinkage will be experienced. A smart processor is a general term which describes that smart devices have been connected or built-into a plant. When this processor is empowered to be used as a gate for every cycle operation, then its reliability will also affect the combined reliability of the plant in every cycle.
Nonetheless there are occasions that a smart processor may, at the same time, improve reliability in other aspects. Three aspects can produce this result: namely, processor injecting immunity power to the plant; processor operating standby redundancy; and processor replacing less reliable hardware control. The optimality of the second aspect relating to standby will be discussed further in the next section of cold-standby.

4 Cold Standby Intrinsic Imperfection

Standby redundancy provides back-up to an operational node. When the main component of the node fails, the back-up component can resume or maintain the node operation. Hence the node will fail only if both the main component and its back-up fail concurrently. With failure + reliability = 1, the combined reliability is given by:

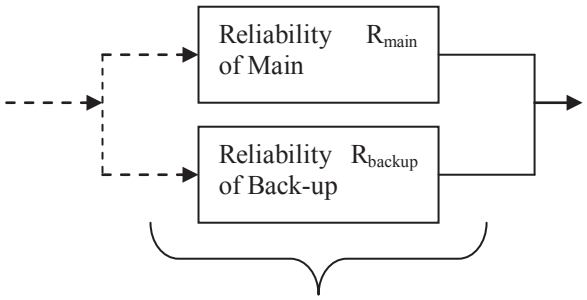

Figure 3. Redundancy Reliability

$$F_{final} = F_{main} \times F_{back\text{-}up} \quad (2)$$

$$R_{final} = 1 - F_{final} \quad (3)$$

$$= 1 - (F_{main} \times F_{back\text{-}up})$$

$$= 1 - (1 - R_{main}) \times (1 - R_{back\text{-}up}) \quad (4)$$

There are three types of standby redundancy: cold standby; warm standby and hot standby. Whilst warm and hot standby provides 'uninterruptible" or "seamless' transfer between the main and standby, yet the cost of wear and tear in the standby may be substantial. That effect is somehow beyond the consideration of this paper.
Smart technology enables powerful sensor and actuation in cold standby. At first sight, the smartness provides redundancy and hence enhances reliability. Yet this paper now explains that the enhancement can never be fully acquired.

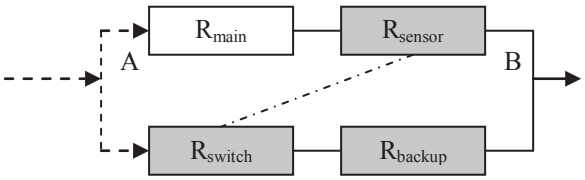

Figure 4. Cold Standby Sensor & Switch

The sensor detects failure of the main; and command the switch to connect the backup for operation. Agarwal and Sahani [2] assumed stochastic behaviour of cold-standby and proposed difference-differential equations to calculate mean-time-to-failure, with assumption reliabilities being exponential time distributions. By using equations (1) and (4),

The figure 4 shows a simpler model to understand the imperfection and limitation of cold-standby. The sensor and the switch are in series with the back-up component. Hence the reliability of the back-up is not by the component alone, but takes sensor and switch into account. The combined reliability of the whole operational node is redundancy reliability between the white box and the grey boxes.

$$R_{back\text{-}up\ combined} = R_{sensor} \times R_{switch} \times R_{backup} \quad (5)$$

$$R_{final} = 1-(1-R_{main})x(1-R_{back\text{-}up\ combined}) \quad (6)$$

There is another constraint in standby arrangement. It is called a "short circuit failure" where the main fails in such a way that it "permanently" short-circuits the operational node between the points A and B. As a result, the back-up cannot be connected to serve redundancy. Mathematically this part of main unreliability cannot be supported by the back-up.
In a smart grid, where many power grids and micro-grids are interconnected, the provision of self-healing is common. Despite the model deliberated does not fully demonstrate the reliability effect because of the mesh restructure and possibility of reverse-flow during self-healing, yet the philosophy is similar to aforementioned. The use of PWM to monitor and to understand and to predict the characteristics of "energy" flow is common. The traditional fault analysis may not be sufficient to understand the ever-expanding complex circumstances that interlinks many major grids and micro-grids.

5 Crossroad and Roundabout Jeopardy

Smart technology is able to enlarge features. The smart phone serves a good example. There are upgrades every month. But undeniably most upgrades utilize the same screen and pad. An appropriate analogy of the scenario is new traffic roads added to junction existing roads, and each time the junction permits one road only to utilize it. Whilst the personal computer sector did provide a multi-tasking breakthrough when Microsoft window was replacing the old DOS command, yet the window is still handicapped in having the machine-human interface working for all tasks and features at the same time. The Figure 5(a) shows how the multiple features utilize the same path. By analogy of crossroad traffic, it is perceived that the ineffectiveness of feature B in the junction may: i) trim down traffic of feature A; or ii) temporary shutdown performance feature A; or iii) halt operation of feature A fully until a reboot of the plant.

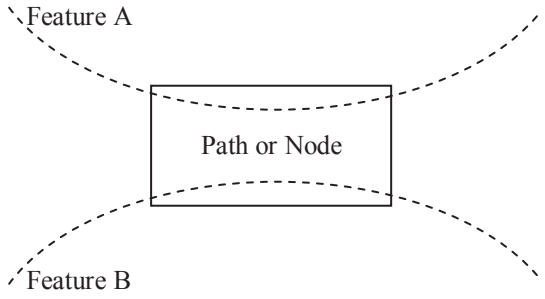

Figure 5(a). Two Features in Same Path

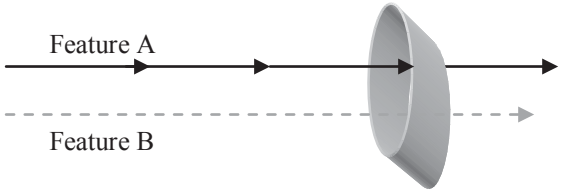

Figure 5(b). Two Features in Coupling

The Figure 5(b) points out another scenario that two features not utilizing the same path may still produce interference. There is coupling effect. When the interference distorts performance to deviate from desirable norm, it is unreliability. Few processes are perfectly sealed against external disturbance as well as it output pollution to external. Processes in vicinity may not standalone and their emissions do propagate to neighbors in each other. The coupling effect is like a roundabout where traffic enters and exits has to cope with neighbors.

6 Software Unreliability

Humans have been paying attention to software unreliability for almost half a century [3][4]. Yet so far there is no "just the thing" model that can be applied to determine trustworthiness of software performance. Researchers like M. R. Lyu commented that software was invisible, and its invisible nature made it both beneficial and harmful [4]. I. Eusgeld et al. presented a good digest of software reliability model [5], yet considered Black Box Reliability Analysis based on failure observations from testing or operation was still a major class of assessment. Unlike hardware reliability assessment which could be done by measuring metrics, software reliability appraisal could not define metrics that could be measured. Often the software risk is concealed, and only at occurrence of turmoil may an engineer and a manager be aware of its existence. But then it has been too late.

Smartness improves performance of many plants and systems by replacing operating parts from hardware to software; and by substituting control parts from hardware to software. The control is a bigger issue to be addressed. A substantial portion of traffic accidents was caused by failure of software in human brains rather than hardware embedded in the car machine.

Software defects are designed faults [5], and they are "unnatural". I. Eusgeld et al. suggested embracing both development and implementation into such design contemplation. As defects are inherent [4] and unknown, test and commissioning based on existing knowledge-base may not be able to detect their jeopardy.

Despite the concern on software reliability, yet it becomes a universal truth and a global commitment to implement more software into new servicing plants and man-machine interfaces. The introduction of smartness from head to toes is an inevitable trend. Smart load, smart phone, smart meter, smart home, smart building, smart grid, and smart city, etc. fill our daily lives.

No plant or system is smart unless it is software driven or processed. No one is certain whether safety is perfect in his system. A classic example of software uncertainty was the "millennium bug". It was a global hot concern about 15 years ago. Notwithstanding human beings, computers and airplanes safely transited to a new century, yet even one second before the clock, no one could be perfectly certain on the scenarios of the next second. There had been much software built to guard against the bug, but the battle was difficult. The engineers were aware of that they were not fighting against an external enemy, but a bug intrinsic in the old software. If the enemy was external, they could build a shield to obstruct the disturbance. That time the enemy was invisible and its DNA was unknown. An extensive effort was made on mitigation of hazard, instead of developing a vaccine to immune the computer. The smooth transition at millennium is still a myth. We were pleased that the bug after all might not always be harmful, but the experience alerted us that software bugs were design faults and concealed.

Of course, software may also improve reliability. 1. As aforementioned, when software may form a protective shield against disturbance and attack, then the chance of hardware failure will be lowered. 2. When the software is to immune or to medicate a plant, then the plant life will become healthier. Where software of other nature is going to be implemented, these two points should also be used as a counter-check on whether the new software would produce the opposite and negative effects. Software attacks are common nowadays. A single virus may paralyze one whole system.

7 Cyber Vulnerability

Open system [4] and open protocol have become norm of new smart systems. Notwithstanding software becomes embedded dependable in its systems, software and minds are now shared and networked. This is an inevitable trend as the benefits of this norm are plenty regardless its scope is beyond the discussion of this paper.

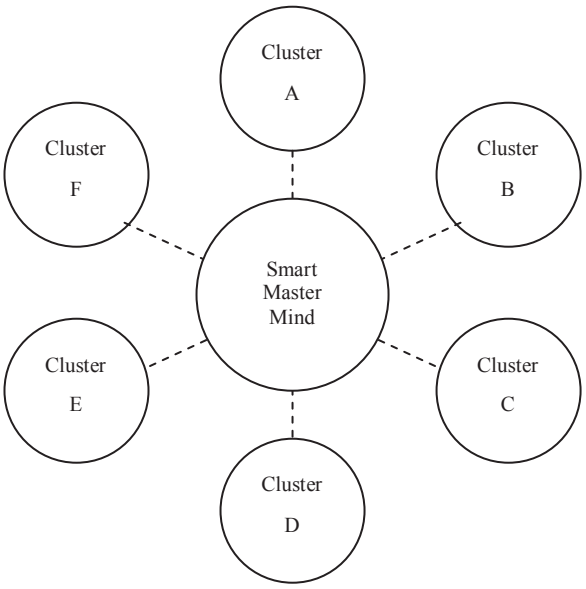

Figure 6

Smart drivers of plants can be viewed as their master minds. Since they are so smart, and therefore human beings attempt to link master minds of plants to form a cluster of master minds; and to connect the many clusters into a system, and eventually an internet universe. There are hierarchies that exhibit structure of feudalism and some others exhibit structure of the united nation. Regardless which structure and to which degree of resemblance, there are two characteristics, some master minds are empowered to be more superior than others; and reliabilities of plants are linked and possibly dependable to each other. There are master minds which drive systems instead of their corresponding plants alone. Many master minds nowadays are allowed to drive more than one system, with their executive arms reaching scores of systems simultaneously. Despite rules of priority are organized, clashes and conflicts are not uncommon. These systems are either homogenous (e.g. power grids) or non-homogenous (computer apps). Most of the master minds nowadays are bilateral directional. But when nerves of the master mind being paralyzed in any system, it is probable nowadays that the other systems will also be halted unless isolation of the faulty system and override/changeover are quickly responsive. For this reason, nations are highly alert about terrorist attacks on networks and grids. One blast may setback the whole world.

The cost of preventing cyber attack by SCADA is as huge as USD7.25 B as reported by Jeff St. John [6]. There have been researches in extensive manner going alongside with the smart grid development. Adam Hahn et al proposed frameworks to evaluate the exposure of cyber attack risk [7], Shan Liu et al worked on vulnerability with particular attention to switching attacks [8]. There have been works in the University of Hong Kong on providing solutions so that smart grids and smart cities are increasingly invulnerable. It will continue to be a hot research issue for some years, and collaboration for it will also be "networked" for concerted effort.

8 Conclusions

The trend of smartness is inevitable, and standalone smartness will be overwhelmed by network smartness. It is envisaged that smartness enhances performance, but may not always enhance reliability at the same time. Reliability is a separate yet important issue to be jointly considered. This paper identifies five major areas of reliability concerns in modern smartness, namely, series shrinkage; cold standby nature; crossroad jeopardy; software reliability and cyber vulnerability. A program to enhance reliability is recommended in each smart system design so that smartness initiation will not amplify failure. Lacking reliability improvement is a threat in modern society which is demanding higher degree of smartness incessantly. In view of it, the paper advocates for concerted effort among all nations for future works on the smart network reliability.

References

[1] U. D. Kumar et al (2000): 'Reliability, Maintenance & Logistic Support, a lifecycle approach'. Kluwer Academic Publishers. Chapters 3 and 4.

[2] S.C. Agarwal1, Mamta Sahani & Shikha Bansal (2010): 'Reliability Characteristic of Cold-standby Redundant System' in IJRRAS 3 (2) May 2010 pp. 193-199.

[3] Jiantao Pan (1999): 'Software Reliability', Dependable Embedded Systems, Carnegie Mellon University, Spring 1999, 18-849b

[4] Michael R. Lyu (2007): 'Software Reliability Engineering: A Roadmap', Proceeding FOSE '07 Future of Software Engineering, 2007, IEEE Computer Society, pp. 153-170.

[5] I. Eusgeld, F.C. Freiling, and R. Reussner (Eds.) (2008): 'Software Reliability", Dependability Metrics, LNCS 4909, pp. 104–125, Springer-Verlag Berlin Heidelberg 2008

[6] Jeff St. John: April 17, 2013: Report: US Smart Grid Cybersecurity, http://www.greentechmedia.com/articles/read/report-u.s.-smart-grid-cybersecurity-spending-to-reach-7.25b-by-2020

[7] Adam Hahn & M.animaran Govindarasu (2011) "Cyber Attack Exposure Evaluation Framework for Smart Grid." IEEE transaction on Smart Grid, Vol 2, No. 4, 2011

[8] Shan Liu, Salman Mashayekh, Deepa Kundur, Takis Zourntos and Karen L. Butler-Purr (2012): A Smart Grid Vulnerability Analysis Framework for Coordinated Variable Structure Switching Attacks Proc. IEEE Power & Energy Society General Meeting, San Diego, California, 2012