Naval Research Laboratory

Washington, DC 20375-5320

NRL/MR/5540--09-9198

Double Rail Tests

GERARD ALLWEIN
IRA S. MOSKOWITZ

Center for High Assurance Computer Systems
Information Technology Division

July 24, 2009

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)*	**2. REPORT TYPE**	**3. DATES COVERED** *(From - To)*
24-07-2009	Memorandum Report	Jan 2008 – Dec 2008

4. TITLE AND SUBTITLE

Double Rail Tests

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER
0601153N

6. AUTHOR(S)

Gerard Allwein and Ira S. Moskowitz

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER
8966

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Naval Research Laboratory
4555 Overlook Avenue, SW
Washington, DC 20375-5320

8. PERFORMING ORGANIZATION REPORT NUMBER

NRL/MR/5540--09-9198

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Office of Naval Research
One Liberty Center
875 North Randolph Street, Suite 1425
Arlington, VA 22203-1995

10. SPONSOR / MONITOR'S ACRONYM(S)

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

12. DISTRIBUTION / AVAILABILITY STATEMENT

Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

We present algebraic operators useful in constructing models for software engineering applied to reliability and security. Double rail testing is a mathematical formalism for analyzing testing situations that have both false positives and false negatives, as well as true positives and true negatives. Furthermore, tests are qualitatively modeled via channel theory, and their quantitative behavior is described as a Shannon binary communication channel. Tests, viewed strictly quantitatively, form a domain (domain theory) and the domain order is determined by the probability of error for tests. The language for tests includes operators for convex sum, sequential (Markov) composition, parallel conjunction and parallel disjunction, and an involution.

15. SUBJECT TERMS
Reliability
Software metrics
Shannon channels
Algebra of tests

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON**
					Gerard Allwein
a. REPORT	**b. ABSTRACT**	**c. THIS PAGE**	UL	12	**19b. TELEPHONE NUMBER** *(include area code)*
Unclassified	Unclassified	Unclassified			(202) 404-3748

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

Double Rail Tests

Gerard Allwein & Ira S. Moskowitz
Center for High Assurance Computer Systems, Code 5540
Naval Research Laboratory
Washington, DC 20375 USA

Abstract

We present algebraic operators useful in constructing models for software engineering applied to reliability and security. Double rail testing is a mathematical formalism for analyzing testing situations that have both false positives and false negatives, as well as true positives and true negatives. Furthermore, tests are qualitatively modeled via channel theory, and their quantitive behavior is described as a Shannon binary communication channel. Tests, viewed strictly quantitatively, form a domain (domain theory) and the domain order is determined by the probability of error for tests. The language for tests includes operators for convex sum, sequential (Markov) composition, parallel conjunction and parallel disjunction, and an involution.

Keywords: reliability, software metrics, Shannon channels, algebra of tests.

1 Introduction

In [6], Shannon's work in reliability [5] was applied for the first time in applications for high assurance systems. The current paper generalizes and extends [6] with an algebra that can be used for software engineering metrics. In some areas of realtime control systems, probability analysis must be substituted for more precise analytical models. The systems can easily become too complicated for the direct analytical approach. Another area that could be addressed using our techniques is hardware-software co-design where the interaction between hardware and software is sometimes not well understood. The complexity issue comes up again in large systems where the system is so complex that a probabilistic model must be used to determine system behavior. The algebra of tests presented in the current paper is intended for use in constructing probabilistic models of systems.

A test is considered to be a channel in the sense of information flow, but what kind of channel is it? It cannot be strictly a communication channel since the objects moving through the channel are objects to be tested, not simply symbols. The objects could be elements which must be managed for secure information flow. In addition, a test may also modify the objects. Hence, certain testing operations might be quite expensive since one cannot test the same object twice. To capture this level of complexity, a more intricate notion of channel is required. This notion is supplied by Barwise and Seligman's channel theory [1].

The framework of channel theory, and the quantitative algebra of tests in this paper, can be used in performing risk analysis. Various kinds of assurance can be enhanced by the repetition of tests. However, what is also important is to measure how much of an increase will result from the expense of repeating tests. Repeating tests is not a particularly sophisticated notion and can easily result in poorer performance than if the test were not repeated. The configuration of how tests are combined matters significantly, and this is formalized in the algebra. Another way to view the algebra for a network of tests is as a formal method for composition of tests via a network of subtests, which is the view in this paper. The analysis and algebra in this paper will support formal tools.

A basic reference for the some of the quantitative mathematics in this paper is a classic paper by Moore and Shannon [5]. This mathematics is augmented by the domain theory contained in [4].

1.1 Formalism

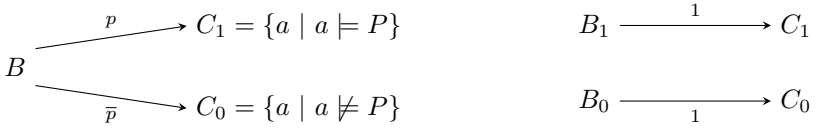
The goal of a test is to classify a collection of objects with respect to some criteria. Let us presume there is a predicate (in the Fraktur font) $\mathfrak{P}(x)$ which, when given an object, determines whether the object has a

Manuscript approved June 24, 2009.

particular property. $\mathfrak{P}(x)$ is much too ideal or perfect to be considered a test. Without confusion, let the property associated with $\mathfrak{P}(x)$ be called $\mathfrak{P}$. x satisfies $\mathfrak{P}$ is denoted $x \models \mathfrak{P}$. At this level of discourse, x is a variable and $\mathfrak{P}$ is an indeterminate. $\mathfrak{P}$ is an element of a Platonic world that is always accurate and never makes a mistake. Most of us live in a non-Platonic world and we must evaluate $\mathfrak{P}(x)$ via a test, $\mathcal{M}$. The test $\mathcal{M}$ might consist of a fair amount of preparation using chemicals, apparatus, etc. Consequently, the actual predicate embodied by a test is something other than $\mathfrak{P}$. Let the predicate embodied by a test be denoted by (the Roman font) P. x satisfying P is denoted as $x \models P$.

One might initially consider that to test for $\mathfrak{P}$ might mean to "prove" of an object x to be tested that x satisfies the conditional $\mathfrak{P} \rightarrow P$, i.e., that if $\mathfrak{P}$ is true of x, then P must be true of x. However, if $\mathfrak{P}$ is not true of x, there is something a bit odd if it were also the case that P were true of x. The test is not doing the correct classifying even though the conditional $\mathfrak{P} \rightarrow P$ holds of x. Tests are expected to classify correctly with respect to $\mathfrak{P}$ and the negation of $\mathfrak{P}$. The term "double rail" comes from CMOS circuits, and refers to the notion that both the positive and negative of a predicate are to be managed.

If $\mathcal{M}$ were a test in a programming language, every state (every x) will force either $x \models P$ or $x \not\models P$, and P is truth functionally equivalent to $\mathfrak{P}$. This kind of test might be pictured via the diagram on the left (below)

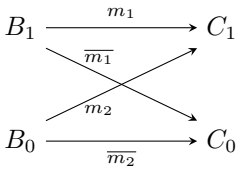


where the p and the $\overline{p}$ represent the proportion of objects of a bin B that the test sorts or classifies either into bin C_0 or into bin C_1. Also, assuming classical probability $\overline{p} = 1 - p$. If P were accurate with respect to $\mathfrak{P}$, then $x \models P$ iff $x \models \mathfrak{P}$.

As a step towards a representation, consider that from a Platonic viewpoint, B is already classified into two bins, say B_0 and B_1 where $B_0 = \{a \mid a \not\models \mathfrak{P}\}$ and $B_1 = \{a \mid a \models \mathfrak{P}\}$. The diagram we would be tempted to draw if P were accurate for $\mathfrak{P}$ is the diagram on the right (above).

There is a probability distribution for just the (B_0, B_1); let the probability distribution associated to (B_0, B_1) be $(p, \overline{p})$ where p represents the proportion of elements that pass the test. This is not represented in the diagram. Thinking of (B_0, B_1) as a set variable, $(p, \overline{p})$ becomes a probability distribution variable and the diagram is accurate as long as the test P is accurate, i.e., it moves elements of B_0 to C_0 with probability 1 and elements of probability B_1 to C_1 with probability 1.

Tests in the real world are rarely as antiseptic. False negatives and false positives are a problem that must be handled when dealing with these kinds of tests. In the case above, elements of B_0 will find their way to C_1 when the test gives a false positive and from B_1 to C_0 for a false negative. Probabilities are associated with these false results. This may be diagrammed as:



where the curious labeling of m_2 for the probability of a false positive and $\overline{m_2}$ for an accurate negative is for simplifying the mathematics in the sequel. This diagram represents only the probabilistic nature of the test. It does not explain details about what the test is for, how the test is performed, or allow us to compare the workings of two different tests except for their probabilistic behavior.

Consider the diagram in terms of B, B_0, and B_1 and observe that $B = B_0 \cup B_1$ and $B_0 \cap B_1 = \emptyset$. Recall B_0 represents elements that truly do not satisfy $\mathfrak{P}$ and B_1 represents elements that truly do satisfy $\mathfrak{P}$. Now consider how the test $\mathcal{M}$ sorts B into bins C_1 and C_0. Let $a \in B$, the argument proceeds by cases since, from the conditions on B, $a \in B_0$ or $a \in B_1$ but $a \notin B_0 \cap B_1$. Let $a \models \mathfrak{P}$ and further let $a \models P$, then P is accurate for a and throws a into bin C_1. Suppose $a \not\models P$, then P is not accurate for a and throws a into bin C_0. By sufficient "runs", a probability m_1 can be assigned to P for accurately sifting elements of B_1 in

which case $1 - m_1 = \overline{m_1}$ is the probability P is inaccurate for sifting elements of B_1. A similar argument holds for bin B_0 and assigning m_2 for false negatives $1 - m_2 = \overline{m_2}$ for true negatives.

A feature of representing tests in the above manner is that a test *labels* the objects it tests. This is important in further classifying objects undergoing testing. The Platonic device of dividing the original bin B into B_0 and B_1 is for the care and feeding of composing tests.

The information about the probabilistic properties of the test P is represented with the following *row stochastic matrix* (RSM):

$$\langle m_1,\ m_2\rangle \stackrel{def}{=} \begin{bmatrix} m_1 & \overline{m_1} \\ m_2 & \overline{m_2} \end{bmatrix} = \begin{bmatrix} \mathsf{K}(C_1|B_1) & \mathsf{K}(C_0|B_1) \\ \mathsf{K}(C_1|B_0) & \mathsf{K}(C_0|B_0) \end{bmatrix}$$

where K is the (Kolmogorov) probability of an event associated with the set, e.g., satisfying $\mathfrak{P}$ is associated with B_1. The set of tests is in bijective correspondence with the set of 2×2 RSMs . The input probability of the bins, represented with $(p, \overline{p})$, is operated on by the RSM to yield another probability distribution:

$$[\,p \quad \overline{p}\,] \circ \begin{bmatrix} m_1 & \overline{m_1} \\ m_2 & \overline{m_2} \end{bmatrix} = \langle pm_1 + \overline{p}m_2,\ p\overline{m_1} + \overline{p}\,\overline{m_2}\rangle$$

with $(p, \overline{p})$ represented by the 2-vector $[\,p \quad \overline{p}\,]$. The output vector of the output represents the probabilistic behavior of test P given its input.

The qualitative behavior of a test is represented by the channel theoretic account in the Appendix where the Gentzen sequents are of the form $\mathfrak{P} \Vdash P$. An entire channel contains a lot of information, however, the rest of this paper is concerned with quantitative behavior. In the sequel, when there is no confusion, a test referred to as $m = \langle m_1, m_2\rangle$ really stands for all that is included in a test; a different test would be denoted $n = \langle n_1, n_2\rangle$.

2 Connecting Tests

The Appendix shows how a test is conceived as a channel of channel theory. Connecting tests via parallel conjunction and/or parallel disjunction may require that objects to be tested be cloned or divided, say, in the way a vial of blood may be divided into two vials with half as much in each. The mechanisms of channel theory do not require this since one could use the identity relation in the channel if the objects for testing require no cloning.

The operations of parallel conjunction, parallel disjunction, and involution of this section indicate functors on the category of test channels. In the succeeding section, the operations of fusion and convex sum also indicate functors. Space restrictions prevent us from exploring this here. These operations (except for convex sum) appear to be similar in spirit (but different in details) from the Dialectica interpretation of linear logic [2].

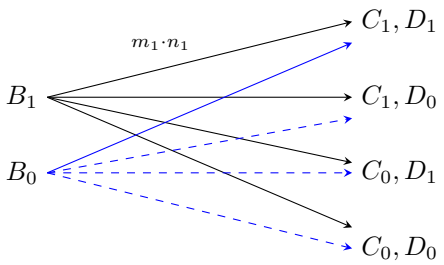
One could also add a further operation, "best of" as in "best 3 out of 5. An object passes the test if it passes the best 3 out of 5 repetitions of the test. These kinds of prescriptions are not always possible given the nature of a test however when present, represent a best case scenario [5]. Space prevents us from addressing these kinds of connections here.

2.1 Parallel Conjunction

Consider two tests where the expectation is that an element satisfies the combined test when and only when it satisfies both of the tests separately. The feel of this test is that an element of an input bin is being tested simultaneously by both tests. This conjunction can be expressed in English by calling it a *parallel conjunction*.

What does it mean to pass both tests? If something starts as satisfying $\mathfrak{P}$, it must satisfy P for both tests, this is simple—no failure. However, if something does not satisfy $\mathfrak{P}$, it is considered to not satisfy P

if it is recognized by as such by at least one of the tests. One can picture parallel conjunction with

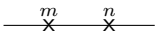


where B_i, C_j are as before and D_i refers the output bins of a test with output predicate Q on the same input and $\langle n_1, n_2 \rangle$ its quantitative behavior. The test results are "vectorized". The first test is the first entry in the 2-vector, and the second test is the second entry in the 2-vector. The line from B_1 to C_1, D_1 has probability $m_1 \cdot n_1$; this probability represents the only correct way for an element from B_1 to test as C_1, D_1, it must pass both tests. Therefore the upper left hand entry of $m \cdot n$ should be $m_1 \cdot n_1$.

What about an element from B_0? As long as it does not pass both tests, it is still considered a failure. Intuitively, if one took a medical test and only passed one test, would the person be confident that they did not have a condition? The failure of test $m \cdot n$ can be seen by following the paths of the three dashed arrows (above) $B_0 \to (C_1, D_0)$, $B_0 \to (C_0, D_1)$, and $B_0 \to (C_0, D_0)$. The sum of the probabilities along these three dashed paths are $m_2 \cdot \overline{n_2} + \overline{m_2} \cdot n_2 + \overline{m_2} \cdot \overline{n_2} = 1 - m_2 \cdot n_2$. Therefore the lower right hand entry of $m \cdot n$ should be $1 - m_2 \cdot n_2$. Collecting together the conditions leads to:

Definition 2.1.1 *The* parallel conjunction *of two tests m and n is defined on the left and represented on the right (below):*

$$m \cdot n \stackrel{def}{=} \langle m_1 \cdot n_1,\ m_2 \cdot n_2 \rangle$$



It is tempting to think of elements to be tested as "flowing" from left to right. This is misleading. The language only shows logical configuration of tests, it is not a flow diagram. Note that $m \cdot m = \langle (m_1)^2,\ (m_2)^2 \rangle$.

2.2 Parallel Disjunction

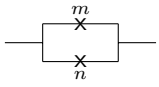
Consider two tests where the expectation is that an element satisfies the combined test just when it satisfies either of the tests. The feel of this test is that an element of an input bin is being tested simultaneously by both tests and success with either constitutes success with the test. This disjunction can be expressed in English by calling it a *parallel disjunction*. One would use this when testing for two disparate properties. As above we "vectorize" the test outputs.

For an element of B_1 going to C_0, both tests must fail; this probability is $\overline{m_1} \cdot \overline{n_1}$. Therefore, the probability of B_1 going to B_1 is simply $\overline{\overline{m_1} \cdot \overline{n_1}}$.

The probability of 1 going to 1, which is (1,1) is as above and is $m_2 \cdot n_2$. Therefore, the probability of 1 going to 0 is $\overline{\overline{m_2} \cdot \overline{n_2}}$.

Definition 2.2.1 *The* parallel disjunction *of two tests m and n is on the right and represented on the left (below)*

$$m \parallel n \stackrel{def}{=} \langle \overline{\overline{m_1} \cdot \overline{n_1}},\ \overline{\overline{m_2} \cdot \overline{n_2}} \rangle,$$



2.3 Involution

There is an involution of the matrices:

Definition 2.3.1

$$\sim \langle m_1, m_2 \rangle \stackrel{def}{=} \langle \overline{m_2}, \overline{m_1} \rangle.$$

Clearly, $\sim \sim m = m$. This corresponds to swapping satisfying the test with not satisfying the test.

Lemma 2.3.2

$$m_2 \leq n_2 \leq n_1 \leq m_1 \text{ implies } \overline{m_1} \leq \overline{n_1} \leq \overline{n_2} \leq \overline{m_2}.$$

Theorem 2.3.3

$$m \parallel n = \sim(\sim m \cdot \sim n)$$

where $\sim$ binds more tightly than $\cdot$ in algebraic expressions.

3 Partial Order on Tests

A partial order on tests can be defined from the notion of probability of error.

3.1 Probability of Error

The probability that a test is wrong depends on the initial probabilities (associated events are disjoint) B_0 and B_1. A test is wrong if it misclassifies an input. So the probability that the test is wrong is:

$$\mathsf{K}(test \;\; wrong) = \mathsf{K}(C_0|B_1)\mathsf{K}(B_1) + \mathsf{K}(C_1|B_0)\mathsf{K}(B_0).$$

The notation is simplified for the test input distribution by setting $p = \mathsf{K}(B_1)$ and $\overline{p} = 1 - p = \mathsf{K}(B_0)$. The random variable describing the test inputs is represented as $(p, \overline{p})$.

Definition 3.1.1 *Let $(p, \overline{p})$ be an input distribution for a test m, then the* **probability of error** *of the test is*

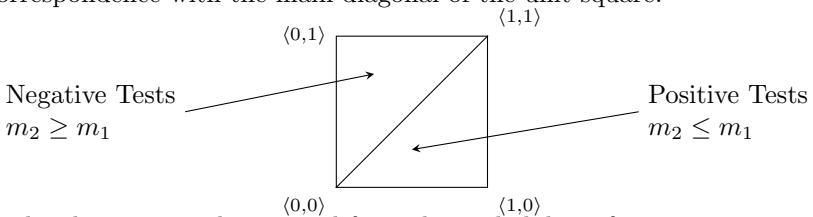
$$e_m(p) \stackrel{def}{=} (p \cdot \overline{m_1}) + (\overline{p} \cdot m_2).$$

Note that if $\overline{m_1} = m_2 = x$, that is the false positive and false negative probabilities are equal, then $e_m(p) = x$.

Let us isolate the reasoning for thinking of a test as being a communication channel. Each element of the bin B can carry a lot of information. However, this information is sorted Platonically by $\mathfrak{P}$ into two bins. In effect, there is only one bit of information one can extract from $\mathfrak{P}$ for each element of B; either $x \models \mathfrak{P}$ or $x \not\models \mathfrak{P}$. To extract other information, we need another predicate. Here, the term "bit" is being used both colloquially and in the Shannon interpretation. The job of the test $\mathcal{M}$ is to transmit information about B. If P were accurate for $\mathfrak{P}$, then P is transmitting precisely the information about the elements of B as reported by $\mathfrak{P}$, i.e., $x \models \mathfrak{P}$ or $x \not\models \mathfrak{P}$. To the extent P fails to be accurate about $\mathfrak{P}$, P fails to transmit accurately information about the elements of B. The source of the transmission is the bin B, the sink is us or whatever is the consumer of what P can say about elements of B.

For the rest of the paper, we restrict ourselves to the subset $\mathbf{N}$ of RSMs that have non-negative determinant. This set $\mathbf{N}$ has been well-studied in [4]. Geometrically $\mathbf{N}$ is the lower right hand triangle of the unit square.

The RSM $\langle 1, \; 0\rangle$ is the identity matrix. The set $\mathbf{D}$ of RSMs of the form $m_d = \langle a, a\rangle, a \in [0, 1]$ are in obvious bijective correspondence with the main diagonal of the unit square.



There is a natural order on tests determined from the probability of error:

Definition 3.1.2

$$m \sqsubseteq n \text{ iff } \forall p(e_m(p) \leq e_n(p)).$$

This definition yields an interval order from domain theory (see [3]). The only part of domain needed in this paper is that the interval order is a partial order.

That the probability of error defines the interval order was observed by Keye Martin and Catuscia Palamidessi (private communication). Interpretations of this order by viewing m and n in their matrix form says that $m \sqsubseteq n$ means m is closer to the identity matrix and n is closer to a matrix with equal rows.

Theorem 3.1.3 (Martin & Palamidessi)

$$m \sqsubseteq n \text{ iff } m_2 \leq n_2 \text{ and } n_1 \leq m_1.$$

The import of this theorem says that one can compare two tests m and n with respect the errors they generate on all input distributions by simply comparing their respective values, i.e., m_1 with n_1 and m_2 with n_2. This relation can be depicted as

It makes sense here to invert the order so that the error decreases as one moves further away from the main diagonal towards $\langle 1, 0\rangle$.

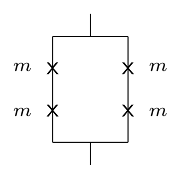
Definition 3.1.4

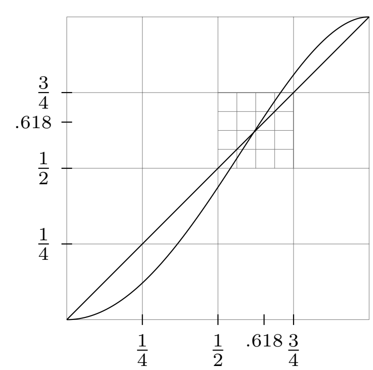
$$m \sqsubseteq_i n \text{ iff } n_2 \leq m_2 \text{ and } m_1 \leq n_1.$$

This partial order is the reverse of the domain order from [4], although itself is not a domain order.

3.2 Networks

Each network is assumed to be made of identical and independent copies of a single test connected in certain ways. The tests are all done simultaneously, much like all coils [5] are energized simultaneously. The network then shows how to combine the test results. Consider the network from [5] now interpreted as a network of test copies of the test m connected as in the diagram on the left:





This network is represented as the test n, where $n = m \cdot m \parallel m \cdot m$. Direct calculations show that $n = \langle 2(m_1)^2 - (m_1)^4, 2(m_2)^2 - (m_2)^4 \rangle$. Is n better than m? Using the partial order, this becomes, does $m \sqsubseteq_i n$ hold? Consider the function $h(x) = 2x^2 - x^4$ which is plotted above (right diagram).

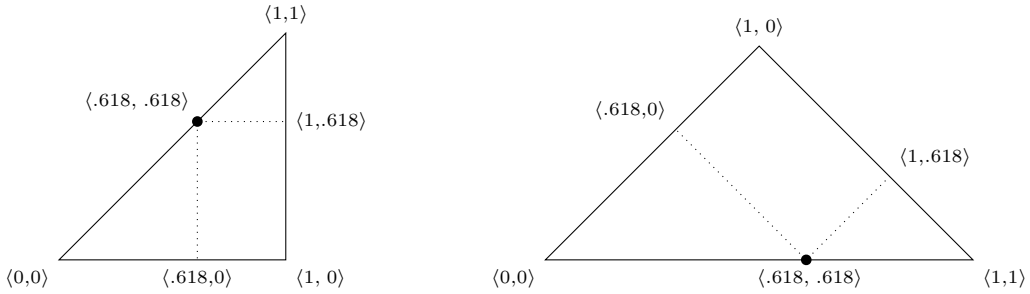
We wish to solve $2x^2 - x^4 = x$, which is equivalent to solving $-x^4 + 2x^2 - x = 0$ (this has four roots, we only care about the ones between 0 and 1). Since $-x^4 + 2x^2 - x = x(1-x)(x^2 + x - 1)$, we see that the solutions are 0,1, and the roots of $x^2 + x - 1$. The root of $x^2 + x - 1$ in the unit interval is $\frac{-1+\sqrt{5}}{2} \approx .618$ (note that $\frac{-1+\sqrt{5}}{2}$ is the multiplicative inverse of the golden mean[1] in and also the golden mean less 1.

If $m_1 \geq \frac{-1+\sqrt{5}}{2}$ and $m_2 \leq \frac{-1+\sqrt{5}}{2}$, then $m \sqsubseteq_i n$. Thus, it is possible to *improve* a test by composition. The 2-dimensional structure for holding m_1 and m_2 shows how to compose tests in order to improve their accuracy. Of course, one may combine tests that are not identical.

[1]We thank Keye Martin for seeing Shannon' s use of .618 in [5] and "knowing" that the inverse of the golden mean had to be a root.

All parallel compositions of tests have the same general form. That is, they look like an S and cross the diagonal exactly once (see [5]) in the open interval $(0, 1)$.

In terms of the order, the diagram on the left (below) shows the region upon which h is guaranteed to iterate channels towards $(0, 1)$.



The region above $\langle\frac{-1+\sqrt{5}}{2}, \frac{-1+\sqrt{5}}{2}\rangle$ in the partial order on tests does form a domain and is pictured with the diagram on the right where the up direction represents an increase in the partial order. Every matrix above $\langle\frac{-1+\sqrt{5}}{2}, \frac{-1+\sqrt{5}}{2}\rangle$ is in the rectangle bordered by $\langle\frac{-1+\sqrt{5}}{2}, \frac{-1+\sqrt{5}}{2}\rangle$, $\langle\frac{-1+\sqrt{5}}{2}, 0\rangle$, $\langle 1, \frac{-1+\sqrt{5}}{2}\rangle$, and $\langle 1, 0\rangle$.

h is strictly increasing in the domain where $\perp = (\frac{-1+\sqrt{5}}{2}, \frac{-1+\sqrt{5}}{2})$ and has the inverse order to the usual interval domain order. Let the extension of h to pairs (determining domain elements) be denoted $\hat{h}$. In this example, $\hat{h}(m) = (2(m_1)^2 - (m_1)^4, (2(m_2)^2 - (m_2)^4)$. $\hat{h}$ is monotone and iteratively increasing above $\perp$ and below $\langle 1, 0\rangle$, i.e., $\hat{h}^i(m) \sqsubseteq \hat{h}^{i+1}(m)$ in the pointwise order. Hence if $m_1 > \pi_1(\perp)$ and $m_2 < \pi_2(\perp)$ (π_i are projections), $h(m_1) > m_1$ and $h(m_2) < m_2$.

This says that if m_2 is below roughly .61 and m_1 is above roughly .61, the network of tests $(m \cdot m) \parallel (m \cdot m)$ is a better test than the test m by itself. Hence, one can make good tests out of mediocre tests.

From [5], the formula h is arrived at via the prescription

$$h(x) = \sum_{i=0}^{n} A_i x^i (1-x)^{n-i}$$

where n is the number of contacts (our tests) in a circuit and A_i is the number of ways a circuit's input can be connect with its output by turning on i contacts and turning off $n - i$. Using the algebra developed in the preceding section,

$$\hat{h}(m) = (m \cdot m) \parallel (m \cdot m) = \langle 1 - (1 - m_1^2)^2, 1 - (1 - m_2^2)^2\rangle.$$

Computing these formulas via the algebra is much easier than attempting to figure out the number of paths through the graphical circuit of tests.

3.3 Trajectories

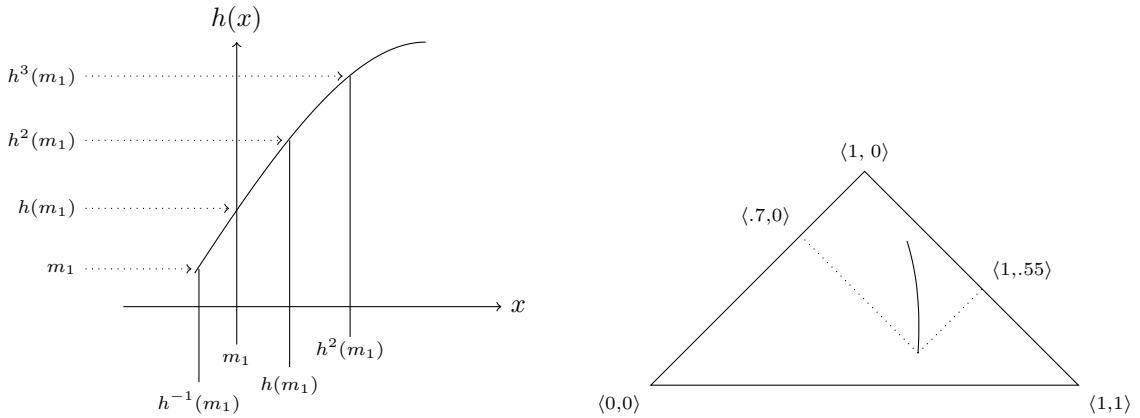
The functions defined by iteration or best-of prescriptions define a trajectory for a test in the partial order. Suppose there is a test with characteristics $m = (m_1, m_2) = (.7, .55)$ and the function $h = 2x^2 - x^4$ from the previous section is applied repeatedly to this test. There will be a sequence of points determined by the iteration, i.e., $\hat{h}(m), \hat{h}^2(m), \hat{h}^3(m)$, etc. These points fall along a parametric curve defined by h.

Consider a $m = \langle m_1, m_2\rangle$ and the curve defined by $h(x) = 2x^2 - x^4$. As long as $m_1 > \frac{-1+\sqrt{5}}{2}$, h will move, via its iterates, m_1 along its curve. Since h is a continuous and strictly increasing curve in the interval $[0, 1]$, it has a unique inverse. Also, the interval $[h^{-1}(m_1), m_1]$ is in the domain of h where h is above the line $y = x$ for $m_1 > \frac{-1+\sqrt{5}}{2}$. Using the scheme from [7] [8], successive arcs from the intervals $[h^i(m_1), h^{i+1}(m_1)]$ can be computed from the arc generated from $[h^{-1}(m_1), m_1)]$ (see diagram on the left below).

The parametric form of the trajectory generated from h is then

$$h^t(m_1) = (t(m_1 - h^{-1}) + h^{-1}(t), h(t)).$$

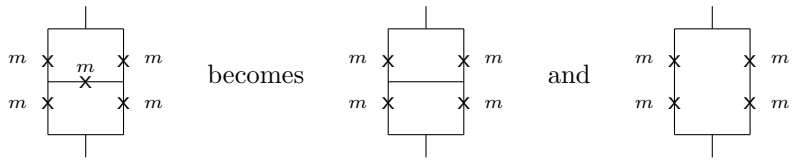
This represents the continuous iteration h^t starting from the fixed position m_1 and t is in the range $[0, +\infty)$. The limit of $h^t(m_1)$ as t approaches $+\infty$ is 1. Using $h^t(m_2)$ for $x < \frac{-1+\sqrt{5}}{2}$ yields a similar analysis and the limit of $h^t(m_2)$ is 0. Combining the two trajectories for $m = (m_1, m_2) = (.7, .55)$ yield a trajectory (see diagram on the right below) in the domain whose bottom point is $\perp = \langle .7, .55 \rangle$ and where $0 \leq t \leq 3$ (t is a real number):



This allows one to take derivatives in the domain where the partial derivative in the increasing m_1 direction and the partial derivative in the decreasing m_2 direction are taken using h^t with respect to t. Thus, the speed at which iteration of a function improves the overall test can be ascertained precisely.

3.4 Convex Sum

The *convex sum* of two tests is used to combine a proportion of one test's output with a proportion of another's. The usual convex sum is $m \oplus_p n \stackrel{def}{=} \langle pm_1 + \overline{p}n_1, pm_2 + \overline{p}n_2 \rangle$. Shannon states that one method of deriving the formula for a network is to pick a contact and replace it twice, once with a short circuit and the resulting network having equation $f(p)$, and once with an open circuit having equation $g(p)$ where p is either m_1 or m_2 when the mesh is constructed with contacts of type m. Hence



Then $f(p) = \overline{\overline{p} \cdot \overline{p} \cdot \overline{p} \cdot \overline{p}}$, $g(p) = \overline{\overline{p \cdot p} \cdot \overline{p \cdot p}}$, and $h(p) = pf(p) + \overline{p}g(p)$. where f is the formula for the network of the middle diagram above and g is the formula for the right hand diagram. This is very close to the convex sum except that the convex sum requires both m_1 and m_2. Since the equation must be computed twice,

$$\hat{h}(m) = \langle m_1 f(m_1) + \overline{m_1} g(m_1), m_2 f(m_2) + \overline{m_2} g(m_2) \rangle.$$

A new operator similar to a convex sum will capture this:

$$n \, \textcircled{m} \, n' = \langle m_1 n_1 + \overline{m_1} n_1', m_2 n_2 + \overline{m_2} n_2' \rangle,$$

Diagrammatically, the test $\stackrel{m}{\text{---}\times\text{---}}$ becomes the connective $\textcircled{m}$ that connects the two diagrams on the right above.

The following theorem is trivially true:

Theorem 3.4.1

$$m \, \textcircled{m} \, m = m.$$

3.5 Serial Conjunction (Fusion) or Sequencing Tests

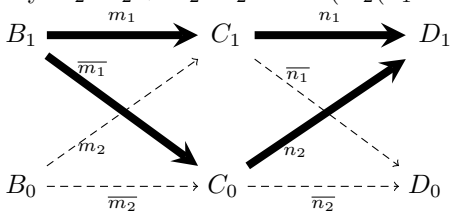
Consider two tests where the bins for the first test's output bins (recall that a test labels the elements by selecting which bin they fall into) become the second test's input bins. This *serial conjunction* or *fusion* gives a notion of sequencing tests.

Definition 3.5.1 *The* serial conjunction *or* fusion *of two tests $m \circ n$ is simply matrix multiplication of their respective behaviors m and n:*

$$m \circ n = \langle m_1,\ m_2\rangle \circ \langle n_1,\ n_2\rangle \stackrel{def}{=} \langle m_1(n_1 - n_2) + n_2,\ m_2(n_1 - n_2) + n_2\rangle.$$

Since fusion is matrix multiplication it is not, in general, commutative.

Referring to the image below, the thick arrows represents the two paths from B_1 to a D_1. The probability that the first path is taken is $m_1 \cdot n_1$, and the probability that the second path is taken is $\overline{m_1} \cdot \overline{n_1} = m_1(n_1 - n_2) + n_2$. Therefore, $m_2 \cdot \overline{n_1} + \overline{m_2} \cdot \overline{n_2} = 1 - (m_2(n_1 - n_2) + n_2)$ is the probability $\mathsf{K}(D_1|B_1)$. Similarly, the dashed arrows below show that path of a B_0 correctly going through the fusion of tests and coming out a D_0; this has probability $m_2 \cdot \overline{n_2} + \overline{m_2} \cdot \overline{n_2} = 1 - (m_2(n_1 - n_2) + n_2)$, which is $\mathsf{K}(D_0|B_0)$.



More information can be extracted from fusion with respect to tests. Let $m = \langle m_1,\ m_2\rangle$. There is a unique line through m that connects $(1, 0)$ to the diagonal. To derive this equation, note that the slope must be negative and is rise over run, hence the slope must be $-m_2/(1 - m_1)$. This gives us

$$y = \frac{-m_2}{1 - m_1}x \quad \text{or} \quad x = -\Big(\frac{1 - m_1}{m_2}\Big)y$$

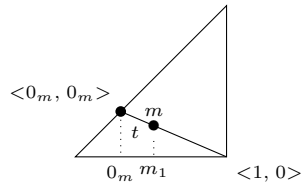
but this must be displaced a bit. Namely, when y is 0, x must be 1, so

$$x = 1 - \Big(\frac{1 - m_1}{m_2}\Big)y.$$

Solving for $x = y$ gives the intersection $< 0_m,\ 0_m >$. Writing out the definition of $m =< m_1,\ m_2 >$ and noting that $\det(m_1, m_2) = m_1 - m_2$, gives us

$$m = \langle m_1,\ m_2\rangle = (1 - t)\langle 0_m, 0_m\rangle + \ t\langle 1, 0\rangle = \langle m_1 = (1 - t)0_m + t, m_2 = (1 - t)0_m\rangle.$$

Hence, $m_1 = m_2 + t$, so $t = m_1 - m_2 = \det(m_1, m_2)$. Since the determinant of m is the length of the interval $[m_2, m_1]$, we let $|m|$ mean $\det(m)$. So the determinant gives us the parameter t. Plugging in $|m|$ for t yields $0_m = \frac{m_2}{1-|m|}$. It helps a bit to see this graphically:



where t is the distance between $\langle 0_m,\ 0_m\rangle$ and m normalized so that the distance from $\langle 0_m,\ 0_m\rangle$ to $\langle 1,\ 0\rangle$ is 1. Hence $m_1 = 0_m + (1 - 0_m)t = 0_m + t - t0_m = (1 - t)0_m + t$, as was computed above.

Let m be a RSM, and let

$$m^n \stackrel{def}{=} \overbrace{m \circ \cdots \circ m}^{n}\ .$$

In more generality, given any sequence of RSMs $m_j, j = 1, \ldots, n$ we have the product $\prod_{j=1}^{n} m_j$ Since m^n and $\prod_{j=1}^{n} m_j$ are well-defined by standard matrix multiplication we may discuss the limit $\lim_{n\to\infty} m^n$ and for an infinite sequence the limit $\lim_{n\to\infty} \prod_{j=1}^{n} m_j$.

Theorem 3.5.2 *The fuse of a test with itself always increases the error*

This is an example of how knowing the domain order can easily determine qualitative characteristics of operations performed on tests.

Theorem 3.5.3 *If* $|m| < 1$ *then* $\lim_{n\to\infty} m^n = \langle 0_m,\ 0_m \rangle$. *If* $|m| = 1$ *then* $\lim_{n\to\infty} m^n = m$.

The proof is a routine induction. Note that if $|m| = -1$, then $m = \langle 0,\ 1 \rangle$ and $\lim_{n\to\infty} m^n$ does not exist because the product m^n oscillates between the identity matrix and $\langle 0,\ 1 \rangle$.

Corollary 3.5.4 *Any test m fuses a distribution* $(x, \overline{x})$ *to* $(0_m, 1 - 0_m)$ *in the limit, i.e.,*

$$(p, \overline{p}) \circ (\lim_{i\to\infty} m^i) = (0_m, 1 - 0_m).$$

4 Conclusion

This paper is the result of viewing a test as a channel of channel theory and then applying some mathematics to extract the quantitative elements of that view. The use of domain theory applies more structure to Shannon's insights by using the probability of error to derive the interval domain on $[0, 1]$. The algebraic operations on the domain are reminiscent of constructs in linear logic, however, the binary relational model of the Dialectica interpretation is inadequate for dealing with tests and the interpretation is unlikely to yield a closed category.

The notion that simply replicating a test does not inherently lead to a better test is at first sight counter-intutive. However, once one attempts to answer the question of the precise relationship between the two instances of the test, the rest of the story becomes formalized in terms of the algebraic operations. These algebraic operations are useful in construction models for testing software which operates in a probabilistic environment such as some types of realtime control systems.

A language for tests would include input predicates and output predicates, and it would also include a graphical language for connecting tests. The graphical language would include "connectives" for all of the algebraic operations dealt with in this paper as well as the "best of" operations.

References

[1] J. Barwise and J. Seligman. *Information Flow: The Logic of Distributed Systems.* Cambridge University Press, 1997. Cambridge Tracts in Theoretical Computer Science 44.

[2] V. C. V. de Paiva. A dialectica-like model of linear logic. In *Proceedings of the Conference on Category Theory and Computer Science, LNCS 389*, pages 341–356. Springer-Verlag, 1989.

[3] G. Gierz, K. Hofmann, K. Keimel, J. Lawson, M. Mislove, and D. S. Scott. *Continuous Lattices and Domains.* Cambridge University Press, 2003. Encyclopedia of Mathematics and its Applications 93.

[4] K. Martin, I. S. Moskowitz, and G. Allwein. Algebraic information theory for binary channels. *Electronic Notes in Theoretical Computer Science*, 158:289–306, 2006.

[5] E. F. Moore and C. E. Shannon. Reliable circuits using less reliable relays: Part I. *Journal of the Franklin Institute*, 262:191–208, 1956.

[6] I. S. Moskowitz and M. Kang. An insecurity flow model. In *Proceedings of the New Security Paradigms Workshop.* ACM, 1997.

[7] M. Ward. Note on the iteration of functions of one variable. *Bulletin of the American Mathematical Society*, 40(10):688–690, 1934.

[8] M. Ward and F. B. Fuller. The continuous iteration of real functions. *Bulletin of the American Mathematical Society*, 42(6):393–396, 1936.