A GENERIC FORMULAIC CHARACTERIZATION OF THE TIME TO REVERSE ENGINEER THE TOLERANCES OF A PRODUCT

Shane K. Curtis  
Research Assistant  
Dept. of Mechanical Engineering  
Brigham Young University  
Provo, Utah 84602  
Email: shanekcurtis@gmail.com

Stephen P. Harston  
PhD Student  
Dept. of Mechanical Engineering  
Brigham Young University  
Provo, Utah 84602  
Email: sharston@gmail.com

Christopher A. Mattson*  
Assistant Professor  
Dept. of Mechanical Engineering  
Brigham Young University  
Provo, Utah 84602  
Email: mattson@byu.edu

ABSTRACT

Reverse engineering is the process of extracting information about a product from the product itself. An estimate of the barrier and time to extract information from any product is useful for the original designer and those reverse engineering, as both are affected by reverse engineering activities. The authors have previously presented a set of metrics and parameters to estimate the barrier and time for product reverse engineering. This work has laid the foundation for the developments of the current paper, which address the issue of tolerance extraction during reverse engineering. Under the developments presented herein, measurement and statistical analysis of the variation between multiple samples of a product are required to reverse engineer its tolerances. When reconstruction is the reason reverse engineering activities are carried out, this level of reverse engineering can be critical, as tolerances ensure that products function properly and consistently. In this paper, we introduce metrics that (i) characterize how the flow of information from a product during reverse engineering changes as additional product samples are evaluated, and (ii) estimate the total barrier and time to reverse engineer the tolerances of a product. Additionally, a simple example is introduced to illustrate how to use the newly developed metrics and to serve as empirical validation.

Keywords: Reverse engineering, barrier to reverse engineering, product imitation, tolerance approximation, time estimation.

NOMENCLATURE

- $B$: Barrier to extract information about a product from the product itself
- $F$: Estimated or actual rate at which information is extracted from a product
- $I$: Electrical current
- $K$: Estimated or actual information remaining in a product
- $L$: Electrical inductance
- $n_K$: Total number of unextracted information levels
- $n_s$: Total number of reverse engineering samples
- $n_T$: Total number of information types
- $P$: Estimated power – effort per time – exerted to extract information contained by a product
- $R$: Electrical resistance
- $S$: A measure of a product’s ability to store information
- $s$: Reverse engineering sample
- $T$: Estimated time to extract information $K$
- $t$: Reference time frame for reverse engineering a product
- $V$: Electrical voltage
- $Z$: Reverse engineering learning lag, a measure of a person’s resistance to change in information flow rate
- $\tau$: Reference time frame when all parameters are known

Subscripts, superscripts, and other indicators

- $[\cdot]^*$ pertains to the product as a whole
- $[\cdot]_s$ pertains to a single reverse engineering sample
- $[\cdot](t)$ is a function of time, in the $t$ domain
- $[\cdot](\tau)$ is a function of time, in the $\tau$ domain
1 Introduction and Literature Survey

In the realm of engineering design, many look at the concept of reverse engineering in a good light, referring to it as product verification, data enhancement or development [1], and competitive benchmarking [2]. On the other hand, opponents to reverse engineering associate it with intellectual theft or piracy [3], and copyright infringement [4]. In each case, reverse engineering is a vital part of the product realization process. Therefore, an understanding of the barrier and time to reverse engineer products is beneficial to those creating new designs and those reverse engineering, as both look to maximize their product realization efficiency [5, 6]. The authors have previously presented a set of metrics and parameters to estimate the barrier and time for product reverse engineering [7]. This has laid the foundation for a new study focusing on geometric and dimensional tolerance extraction. Reverse engineering tolerances requires special attention, as it involves the evaluation of multiple samples of the same product, where each additional sample becomes progressively easier to evaluate due to the knowledge gained from previous samples. In this paper, we add metrics to those previously developed, which can accurately determine the barrier and time to reverse engineer the tolerances of a product.

As stated above, reverse engineering carries multiple disparate connotations in the literature. Therefore, to preserve clarity in the present paper, we provide the following important definitions:

### Reverse Engineering

is the process of extracting information about a product from the product itself. This includes reverse engineering tolerances, which involves extracting sample-to-sample variation from a product population.

### Time to Reverse Engineer

is the total required man-time to reverse engineer a product without consideration to parallel activities.

### Barrier to Reverse Engineering

is anything that impedes reverse engineering.

As defined, reverse engineering tolerances is a subset of reverse engineering. Consequently, the same barriers that make reverse engineering challenging also create difficulties for tolerance extraction. Examples of these barriers may include the convoluted surfaces of an airplane propeller or the critical, pitch curves of non-circular gears. These features make it difficult to reverse engineer product geometry as well as tolerances, but it is important to note here that large barriers do not necessarily imply large times to reverse engineer. Barriers and time to reverse engineer are related, but distinct. A more detailed discussion on barriers to reverse engineering can be found in [7].

Reverse engineering with the intent of reconstructing a product for future manufacture is incomplete until tolerances are allocated to the product. In this paper, we consider tolerances to be an acceptable amount a dimension is allowed to vary. If the dimensions of a product vary more than the allowable tolerances, then the probability of the product failing to assemble or function correctly increases. This will inevitably lead to costly repairs, poor performance, and dissatisfied customers, all of which diminish the product’s effectiveness [8].

Optimally allocating tolerances is a typical, yet challenging task in engineering design. An overview of the many methods used to allocate tolerances when designing a product can be found in [9]. When reverse engineering, the process becomes more difficult [10], as it requires a significant amount of skill and experience to match the original tolerances of a product. This alone can be seen as a barrier to reverse engineering tolerances [11]. As a consequence, various methods have been presented in the literature to help approximate dimensional and geometric tolerances when reverse engineering [12–14]. One method includes using the surface characteristics of a product, such as surface finish or texture, to approximate dimensional tolerances [12]. Additionally, Kaisarlis et al. [13] have developed a knowledge-based, computer-aided tolerance allocation system requiring geometric, production, roughness, and functional data about the product. Their program uses this data to determine possible tolerance ranges. Another method involves performing dimensional analysis on multiple samples of the same product and comparing the results to discover possible manufacturing variations as an aid to establishing tolerances [14]. In the end, a designer must accept or reject these tolerance approximations; the decision will certainly be affected by the quality of the information extracted from the product.

Therefore, the accuracy of extracted information is important. When one product is reverse engineered, the part, in most cases, is just a single member of a distributed population, where variation is undoubtedly present [15]. Furthermore, the precision of measuring devices, such as digital calipers or coordinate measuring machines, introduce uncertainty into extracted product data [16]. As a result, an appropriate statistical analysis needs to be performed in order to test hypotheses on the true nominal values of geometric information in a part. This involves determining an adequate sample size based on a predetermined confidence level and acceptable error [17]. As the number of available parts for the sample increases, so does the accuracy of the reverse engineering data [18]. Thus, when extracting tolerances or dealing with modeling accuracy in reverse engineering, information must be extracted from multiple samples of the same product, which requires a significant amount of time.

Our tenet is that an estimate of this time would benefit both those who initially design products and those who attempt to re-
verse engineer a product. To those designing products, efforts could be made to include more barriers to reverse engineering tolerances in the products that they design. To those reverse engineering, it would be beneficial to have an estimate of how long this process will take so as to know whether it is more advantageous, from a time consumption perspective, to include tolerance extraction in reverse engineering activities.

In this paper, we introduce parameters and metrics that (i) characterize how information flow rates increase with each sample when extracting tolerances, and (ii) predict the total barrier and time to reverse engineer the tolerances of a product. While there are multiple applications of the metrics and relationships presented here, they have been developed with the intention of using them in conjunction with numerical optimization techniques to maximize the barrier and time to reverse engineer a product and its tolerances, so as to make reverse engineering more difficult for competitors.

The remainder of this paper is organized as follows: In Sec. 2 we present a brief overview of the pertinent reverse engineering metrics developed previously by the authors. The additional metrics, capable of estimating the time and barrier of tolerance extraction, are then introduced in Sec. 3, followed by a simple example illustrating the use of the newly developed relationships in Sec. 4. Concluding remarks are provided in Sec. 5.

2 Metrics for Nominal Information Extraction

In this section, we summarize the metrics for evaluating the barrier and time to reverse engineer a product, which were previously presented by Harston and Mattson in [7]. These relationships provide the foundation for the new developments in this paper. Ohm’s law [19] has been adapted to describe the barrier, $B$, to reverse engineering as

$$B = \frac{P}{F^2}$$

where $P$ is the power – effort per time exerted to extract information – and $F$ is the rate at which information is extracted from a product. The value of $P$ is constrained by

$$0 < P \leq 1$$

where zero represents no effort being put forth to reverse engineer a product and one signifies full effort at maximum efficiency. The storage capacity, $S$, of a product is defined as

$$S = \frac{KF}{P}$$

where $K$ is the amount of unextracted information remaining in a product. Using these definitions, the time required to reverse engineer a product (without extracting tolerance information) can be accurately predicted using the following exponential decay relationship

$$T = -BS\ln\left(\frac{K}{K(0)}\right)$$

where $K(0)$ is the amount of information initially stored by the product. Thus, it follows that $K$ is constrained to

$$0 < K \leq K(0)$$

which ensures that Eq. 4 yields a finite quantity of time.

The time predicted by these metrics is useful and accurate when reverse engineering a product once, but if the same product is analyzed again, the metrics need to be modified. With subsequent iterations of the reverse engineering process, knowledge about the product from previous reverse engineering samples is retained and used to decrease the time necessary for information extraction. Therefore, the metrics in this section, if multiplied by the sample size, over estimate the total time to reverse engineer multiple samples of the same product. As some products are repeatedly reverse engineered to improve dimensional accuracy or to determine potential dimensional and geometric tolerances, additional metrics are needed. In the following section, the tolerance metrics are introduced.

3 Metrics for Tolerance Information Extraction

In this section, we develop the metrics for predicting the time and barrier to reverse engineer tolerances. The presentation of the metrics is divided into three main parts. In Sec. 3.1 we examine how the flow rate of information changes during the process of reverse engineering. Then in Sec. 3.2 we consider a simple resistor-inductor circuit, as it can be used to model the change in information flow rate that occurs with successive reverse engineering of samples of the same product. Finally, in Sec. 3.3 we introduce the tolerance metrics.

3.1 The Behavior of Information Flow Rates While Reverse Engineering Tolerances

In the context of reverse engineering, flow rate of information, $F$, is defined as the time derivative of unextracted information remaining in a product, or

$$F = \frac{dK(t)}{dt}$$
Time 
Unextracted Information  ( K )
K 1
K 2
K 3
K 4
K 5
K n

Figure 1. UNEXTRACTED INFORMATION IN A PRODUCT AS A FUNCTION OF TIME. THE CURVES FOR MULTIPLE REVERSE ENGINERING SAMPLES ARE COMPARED.

The flow of information changes with each sample when reverse engineering multiple samples of the same product. This trend is illustrated in Fig. 1, which plots K (unextracted information remaining in a product) as a function of time for several reverse engineering samples. The first curve labeled 1, representative of reverse engineering sample 1, resembles an exponential decaying relationship. The second curve, labeled 2, is steeper than curve 1, which means the flow rates are generally larger. Flow rates continue to increase for subsequent sample curves denoted n. As will be shown, if the reverse engineering sample size is sufficiently large, then the curves asymptotically approach a linear prediction, which is the dashed line marked by the in the plot. The rate at which the extraction of information moves from exponential to linear in nature has a large impact on estimating the time to reverse engineer tolerances.

To more closely examine the sources behind the increase in information flow rates, it becomes necessary to consider F at various values for K on each reverse engineering sample curve. These discrete values are termed unextracted information levels and are determined by the type of information being extracted. For example, in the case of geometric dimension extraction, each unextracted information level would specify how many dimensions have not been extracted. Therefore, flow rate and unextracted information can be expressed in matrix form:

\[ F = \begin{bmatrix} F_{1,1} & F_{1,2} & \cdots & F_{1,n_{i}} \\ F_{2,1} & F_{2,2} & \cdots & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ F_{n_{K},1} & \cdots & \cdots & F_{n_{K},n_{i}} \end{bmatrix} \]  

(7)

\[ \vec{K} = [K_{1}, K_{2}, \ldots, K_{n_{K}}]^T \]  

(8)

where each entry in \( \vec{K} \) is an unextracted information level, \( n_{K} \) indicates the total number of unextracted information levels and \( n_{s} \) is the total number of reverse engineering samples (i.e. the product sample size). The subscripts \( [i,j] \) are used to distinguish unextracted information level, \( i \), and reverse engineering sample, \( j \). The plot in Fig. 2 illustrates how this notation is used by locating where \( F_{3,3} \) is evaluated. Returning to Fig. 1, the horizontal dashed lines help visually track unextracted information levels along different sample curves. As information flow rates are characterized by the slopes in the plot, it is clear that flow rates vary, even at the same unextracted information level for different reverse engineering samples.

The variation in flow rates can be credited to the fact that reverse engineering encompasses more than just measuring dimensions, for example. It includes secondary procedures such...
as deciding which dimensions are pertinent, finding the dimensions in the product, documenting or recording the dimensions on a hand drawing or in a CAD system, and verifying that all the needed dimensions have been extracted. When all of the aforementioned steps are performed, the flow rate of information is low, in comparison to when none or few of the secondary procedures are necessary for information extraction. This implies that the fastest, or largest, flow rate occurs when information is simply extracted without utilizing any secondary procedures.

Harston and Mattson [7] have termed the fastest theoretical flow rate the initial flow rate, or \( F(\tau_0) \), where \( \tau_0 \) refers to when time equals zero in the \( \tau \) reference time frame, in which all parameters are known. In the context of reverse engineering tolerances, \( F(\tau_0) \) is \( F_{1,1} \). This flow rate occurs when the reverse engineering process has achieved maximum efficiency (i.e. \( P = 1 \)). In Fig. 1 the initial flow rate is also the initial slope of each sample curve, or

\[
F_{1,j} = F_{1,1}, \quad \forall \ j \in \{1, 2, \ldots, n_s\} \tag{9}
\]

Therefore, the initial flow rate remains the same for an individual, regardless of reverse engineering sample. This assumes that information requiring less extraction time is extracted from a product first, followed sequentially by units of information requiring more time. This may or may not happen in practice; regardless, when empirical data gathered by the authors is rearranged according to the time to extract each unit of information – with the shortest times placed first – this relationship generally holds true.

Additionally, when a person reverse engineers a product for the second time (\( j = 2 \)), he or she utilizes some of the knowledge gained while reverse engineering the product the first time, obviating some of the steps of the reverse engineering process. For example, if someone is reverse engineering the geometry of a piston for the second time, the pertinent dimensions of the piston have already been determined during the first reverse engineering sample, as well as an appropriate documentation procedure. This is characterized in Fig. 1, where the slopes along curve 2 are generally steeper than those of curve 1, resulting in less total time to reverse engineer the product. Reverse engineering additional samples will yield similar results – flow rates will continue to increase and the reverse engineering time will continue to decrease, until the flow rates at all unextracted information levels \( F_{1,j} \) through \( F_{n_s,j} \) have reached \( F_{1,1} \).

Therefore, when the flow rates of different samples at a particular unextracted information level \( i \) are compared to one another, the following is true

\[
F_{1,1} \leq F_{i,j} \leq F_{i,n_s}, \quad \forall \ i \in \{1, 2, \ldots, n_K\}, \quad \forall \ j \in \{1, 2, \ldots, n_s\} \tag{10}
\]

In other words, the flow rate at a particular unextracted information level is bound by the flow rate of the first reverse engineering sample \( F_{1,1} \) and the flow rate of the last reverse engineering sample \( F_{i,n_s} \). Additionally, if the product sample size is sufficiently large, the curves in the plot approach a linear prediction with the slope of the initial flow rate (the dashed line marked by the \( \infty \) in Fig. 1), or

\[
\lim_{j \to \infty} F_{i,j} = F_{1,1}, \quad \forall \ i \in \{1, 2, \ldots, n_K\} \tag{11}
\]

This suggests that the individual reverse engineering tolerances drives all flow rates toward the initial flow rate, essentially maximizing information extraction efficiency.

The question remains as to how quickly (in terms of reverse engineering samples) information flow rates approach the initial flow rate. Some individuals are slow learners with regards to reverse engineering, while others are not. Therefore a parameter is needed, reflective of a person’s reverse engineering learning capability, to help characterize this behavior. In the next section, an analogous electrical model is introduced to meet this need.

### 3.2 Simple Resistor-Inductor Circuit as a Foundation

In this section, we analyze the simple resistor-inductor circuit shown in Fig. 3 as a foundation for the development of metrics capable of predicting the time and barrier to reverse engineer tolerances. The analysis of the circuit is applicable to reverse engineering tolerances for two reasons: (i) the first order response of electrical current in this circuit behaves like information flow rates when reverse engineering multiple product samples and (ii) the inductance in the circuit provides a comparable parameter that can be adapted to characterize a person’s reverse engineering learning capability.

Inductance is a measure of an object’s ability to store energy in the form of a magnetic field [20] and resisit change in the flow of electrical current [21]. The voltage across an inductor, \( V_L(t) \),

![Figure 3. SIMPLE SERIES RESISTOR-INDUCTOR CIRCUIT.](image)
is

\[ V_L(t) = L \frac{dI(t)}{dt} \]  

where \( L \) is the inductance, and \( I(t) \) is electrical current. Using the above relationship, we can apply Kirchhoff’s voltage law to the resistor-inductor circuit in Fig. 3 to find the governing equation

\[ V(t) = L \frac{dI(t)}{dt} + I(t)R \]  

where \( R \) is resistance and \( V(t) \) is provided by an ideal voltage source.

Solving Eq. 13 for \( I(t) \) yields the following

\[ I(t) = \frac{V_0}{R} e^{-tR/L} + \frac{V_\infty}{R} \left(1 - e^{-tR/L}\right) \]  

where \( V_0 \) is the initial voltage supplied to the circuit, and \( V_\infty \) is treated as a step input being applied when \( t = 0 \). The voltage input and current output are plotted in Fig. 4. As the graphs show, the voltage jumps to \( V_\infty \) at \( t = 0 \), however the current lags the voltage in the circuit due to the inductor. A similar input-output situation occurs when reverse engineering tolerances – we assume an individual gives a full and unchanging effort to reverse engineer each sample, however information flow rates may require multiple reverse engineering samples before they reach the initial flow rate (the theoretically fastest flow rate).

Furthermore, using Ohm’s law \( (V = IR) \), we can rearrange Eq. 14 to get

\[ I(t) = I_0 e^{-tR/L} + I_\infty \left(1 - e^{-tR/L}\right) \]  

where \( I_0 \) is the current initially flowing through the circuit and \( I_\infty \) is the current flowing through the circuit after a long time. This form of the equation is more useful from the perspective of reverse engineering, as electrical current – the flow of electrons – can be compared to the flow of information from a product, which is a relatively easy parameter to measure in reverse engineering.

The ratio of inductance to resistance, \( L/R \), is the time constant for the circuit. This value characterizes how fast the current in the circuit responds to changes in input voltage. Larger values of inductance and lower values of resistance result in larger time constants. By examining the resistor-inductor circuit after the voltage step input has been applied, we can find an equation for \( L \), which can also easily be adapted to reverse engineering.

At any time after the voltage step input \( (t \geq 0) \) Eq. 13 becomes

\[ V_\infty = L \frac{dI(t)}{dt} + I(t)R \]  

After a sufficient amount of time has passed, the current will remain constant, making \( dI(t)/dt = 0 \) and \( I(t) = I_\infty \), or

\[ V_\infty = I_\infty R \]  

By subbing Eq. 17 into Eq. 16 and solving for \( L \) we get

\[ L = \frac{R(I_\infty - I(t))}{dI(t)/dt} \]  

This is not a traditional approach to find \( L \) for most electrical applications; however, the terms in Eq. 18 easily convert to quantifiable parameters and metrics in reverse engineering \( (B, F) \). Therefore, we can use this relationship for \( L \) to characterize a person’s reverse engineering learning capability.
3.3 Metrics for Tolerance Information Extraction

The ability to predict the change in flow rates from sample to sample at different information states is vital to accurately predicting the time to reverse engineer tolerances. The analysis of the resistor-inductor circuit led Eq. 15, which can now be modified to describe an information flow rate at an unextracted information level, \( i \), and reverse engineering sample, \( j \), with the following

\[
F_{i,j} = F_{i,1}e^{-(j-1)B/Z} + F_{1,1}\left(1 - e^{-(j-1)B/Z}\right) \tag{19}
\]

where \( B \) is the barrier to reverse engineering, as defined in Eq. 1, \( Z \) is termed the reverse engineering learning lag, and \( F_{1,1} \) is the theoretically fastest flow rate, also termed the initial flow rate. The flow rate from the first reverse engineering sample at an unextracted information level, \( i \), is denoted by \( F_{i,1} \). This value is determined by solving for \( K \) in Eq. 4, and substituting into Eq. 6, which yields

\[
F_{i,1} = \frac{-K(0)}{BS}e^{-i/BS} \tag{20}
\]

This equation can be further simplified by substituting Eq. 1, 3, and 4 for \( B, S \), and \( t \), respectively, allowing \( F_{i,1} \) to be rewritten as

\[
F_{i,1} = \frac{F_{1,1}K_i}{K_1} \tag{21}
\]

In this form, the first flow rate for each information state can easily be calculated and used in Eq. 19.

The reverse engineering learning lag, \( Z \), is defined by the following

\[
Z = \frac{B(F_{i,1} - F_{i,j})}{dF_{i,j}/ds} \tag{22}
\]

where \( dF_{i,j}/ds \) indicates the change in information flow rate, \( F_{i,j} \), per reverse engineering sample, \( s \), for any flow rate besides the initial flow rate. The reverse engineering learning lag is analogous to inductance (see Eq. 18) in an electrical circuit – both measure resistance to change in flow rates. The reverse engineering learning lag is not the same for all individuals, and therefore needs to be determined through a uniform information extraction test. More information on how this is to be done can be found in Sec. 4.

The effect of the reverse engineering learning lag can be seen in Fig. 5, which illustrates how Eq. 19 predicts flow rates when reverse engineering tolerances. At a particular unextracted information level, \( i \), the flow rate is plotted as a function of reverse engineering sample. The three curves labeled as \( \alpha \), \( \beta \) and \( \gamma \) represent three different individuals, all of whom have a unique reverse engineering learning lag. The flow rate for each individual starts at the value predicted in the first sample, \( F_{i,1} \), and as the samples increase, the flow rates approach the asymptotical limit of \( F_{i,1} \). The ratio of \( Z \) to \( B \) determines how quickly (in terms of samples) the flow rates rise to \( F_{i,1} \). If the ratio is large, many samples will be needed for the flow rate to reach its final state. In the plot, curve \( \gamma \) has the largest ratio of \( Z \) to \( B \); consequently, it rises to \( F_{i,1} \) the slowest. Likewise, curve \( \alpha \) has the lowest ratio and rises the quickest.

With a relationship defined for how the flow rate changes, the total time to reverse engineer tolerances can be calculated with

\[
T = -BS\ln\left(\frac{K_{nk}}{K_1}\right) + \sum_{i=1}^{n_K} \sum_{j=2}^{n_i} \frac{1}{F_{i,j}} \tag{23}
\]

where \( n_K \) is the total number of information states, \( n_i \) is the total number of reverse engineering samples, \( F_{i,j} \) is calculated with Eq. 19, and the 1 in the numerator is one unit of information. The parameters and metrics that make up Eq. 23 can easily be calculated for any individual or product. Thus, the task of accurately estimating the time to reverse engineer a product becomes simple and straightforward. An example of how this is done is also provided in Sec. 4.

The time required for each individual reverse engineering sample (beyond the first) can also be determined by modifying.
Eq. 23 to get

\[
T_j = \sum_{i=1}^{n} \frac{1}{F_{i,j}}
\]  

(24)

where the subscript \( j \) distinguishes reverse engineering sample and the 1 in the numerator signifies one unit of information. It is important to note that the metrics developed here use unextracted information levels defined in Sec. 3.1 to determine the flow rates from the first reverse engineering sample, using Eq. 21. As discrete points are used to characterize the entire curve, approximation error is introduced into the model. Therefore, to maintain a higher degree of accuracy, it is more appropriate to use Eq. 4 for the first reverse engineering sample.

Up until this point, the metrics introduced have not considered the type of information being extracted from a product. A more in depth discussion on the taxonomy of information in products can be found in [7]. Information type is a significant factor in reverse engineering, as the barrier and time for reverse engineering depend on the type of information that is contained in a product. Each information type has a distinct initial flow rate. Therefore, every information type needs to be considered separately. This will result in a different time to reverse engineer tolerances for each information type, denoted \( T_i \). Thus, the total time to reverse engineer the tolerances of a product can be found with the following

\[
T^* = \sum_{i=1}^{n_T} T_i
\]  

(25)

where \( n_T \) is the total number of information types contained in the product.

The barrier to reverse engineer tolerances is the same barrier that has been defined in Eq. 1. Each information type has a unique barrier; however, this barrier does not change with additional reverse engineering samples, despite the fact that information flow rates do increase. This is similar to the resistor in the resistor-inductor circuit – the value of its resistance remains the same, even though the current passing through it can change. Therefore, the effective barrier to reverse engineer tolerance for an entire product can be calculated using the developments presented in [7].

4 Simple Example: Ballpoint Pen

In this section, we calculate the necessary parameters and metrics to predict the time to reverse engineer the dimensional tolerances of a Bic Round Stic®, ballpoint pen and cap, shown in Fig. 6. We then compare the time estimations to data gathered from an experiment, in which an individual actually reverse engineered 30 pens and caps to determine tolerances, so as to illustrate the validity of the metrics in this paper. For this preliminary study, the pen and cap were chosen due to their geometric simplicity. Future research will focus on exploring the bounds of the developed metrics, wherein more complex products will be analyzed.

The initial flow rate, \( F_{1,1} \), represents the quickest rate at which an individual can extract information from a product. We obtain \( F_{1,1} \) experimentally by using a uniform dimension extraction test. The goal of the test is to measure the rate at which a person can extract information from a product when no secondary reverse engineering procedures are performed (see Sec. 3.1). In
the test, an individual is asked to familiarize themselves with a particular dimension on a product. After this is done, the individual receives a measurement tool and the time is then recorded for them to measure the dimension. The process is repeated for many different dimensions of the same information type and the extraction rates are averaged to determine $F_{1,1}$. The resulting $F_{1,1}$ determined by the test can be used to calculate the metrics for any product that contains the appropriate information type. For the individual reverse engineering the tolerances of the ballpoint pens, $F_{1,1}$ was discovered to be 0.065 (dimensions per second).

Power is the measure of the effort being exerted per time by the person reverse engineering the product. The power associated with the initial flow rate is $P_{1,1}$ and is assumed to be 1, meaning the person reverse engineering gave a full and solid effort.

The first unextracted information level, $K_1$, indicates how much information is contained by the product initially. A total of 28 linear dimensions were deemed necessary and pertinent to reverse engineer the pen and cap, making $K_1$ equal to 28 (dimensions). Consequently, $n_K$, the total number of unextracted information levels is also 28.

With $F_{1,1}$, $P_{1,1}$, and $K_1$ defined, the barrier to reverse engineering $B$ and the storage ability $S$ were calculated with Eqs. 1 and 3, respectively. The geometric storage ability of the pen and cap is 1.82. The barrier to reverse engineering is 236.89.

\[ B = \frac{F_{1,1}P_{1,1}K_1}{n_K} \]

\[ S = \frac{1}{B} \]

\[ S = \frac{1}{236.89} \]

\[ S = 0.0042 \]

\[ B = \frac{1}{0.0042} \]

\[ B = 235.71 \]

\[ S = 1.82 \]

\[ B = 236.89 \]
of the ballpoint pens and caps was estimated using Eq. 23. All subsequent sample times were determined using Eq. 24. The time required for the first sample was predicted using Eq. 4.

It is now possible to predict the time to reverse engineer tolerances. The metrics defined in this section are listed in Tab. 1. With these values, it is absolutely consistent and suggest that the metrics presented in this paper accurately predict the time required to reverse engineer tolerances. The actual extraction results along with the time predictions and errors are listed in Tab. 2.

Additionally, the plots in Fig. 7 display the actual results for pen samples 2, 5, 10, and 30. The data in the plots has been rearranged to according to the time to extract each dimension - with the shortest times plotted first - and may not be plotted in the order of dimension extraction. The results in each plot are compared to linear, exponential, and tolerance predictions. As discussed in Sec. 3.1, the slope of the linear prediction is $F_{1,1}$ and the y-intercept is at $K_1$. The exponential and tolerance predictions are determined from Eqs. 4 and 24, respectively. The metrics developed for tolerance extraction provide relatively accurate predictions for information flow rates at each unextracted information level, as well as for the times of each reverse engineering sample. Consequently, the newly developed metrics were able to predict the total time to reverse engineer the tolerances of the pen and cap with an error of 8.39%.

### Table 2. TABLE OF PREDICTED AND ACTUAL TIMES TO EXTRACT GEOMETRIC INFORMATION FROM BALLPOINT PENS AND CAPS. TIME IS IN SECONDS.

<table>
<thead>
<tr>
<th>Sample #</th>
<th>Time (Actual)</th>
<th>Time (Prediction)</th>
<th>% Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2,017</td>
<td>1,431</td>
<td>-29.08</td>
</tr>
<tr>
<td>2</td>
<td>799</td>
<td>946</td>
<td>18.40</td>
</tr>
<tr>
<td>5</td>
<td>606</td>
<td>619</td>
<td>2.20</td>
</tr>
<tr>
<td>10</td>
<td>519</td>
<td>498</td>
<td>-4.18</td>
</tr>
<tr>
<td>30</td>
<td>465</td>
<td>433</td>
<td>-6.90</td>
</tr>
<tr>
<td>TOTAL</td>
<td>17,817</td>
<td>16,322</td>
<td>-8.39</td>
</tr>
</tbody>
</table>

(All Samples)

The reverse engineering learning lag, $Z$, characterizes a person’s resistance to change in information flow rates. It is calculated using Eq. 22, which requires a flow rate other than the initial flow rate, $F_{1,i}$, and its associated derivative with respect to reverse engineering sample, $dF_{i,j}/ds$. These values are determined for an individual experimentally in a uniform dimension extraction test, similar to how $F_{1,1}$ is determined. However, in this test the person must extract a single dimension from multiple product samples, as if they were reverse engineering tolerances. For the individual analyzing the ballpoint pens, $dF_{i,j}/ds$ was found to be .00842 (dimensions per second per sample) when $F_{i,j}$ was .00721 (dimensions per second). With these parameters defined, $Z$ was calculated to be 1625.

The flow rates for the first reverse engineering sample, $F_{1,1}$, for all information levels $i = 1$ through $i = n_K$ were calculated using Eq. 21, but are not shown here to keep the presentation simple. The remaining parameters and metrics that have been defined in this section are listed in Tab. 1. With these values, it is now possible to predict the time to reverse engineer tolerances. The time required for the first sample was predicted using Eq. 4. All subsequent sample times were determined using Eq. 24. The total time to reverse engineer the linear dimensional tolerances of the ballpoint pens and caps was estimated using Eq. 23.

Uninfluenced by the predictive measures, an individual was asked to reverse engineer the tolerances of the pen and cap. This individual reverse engineered 30 pen samples over a period of two days. Efforts were made to avoid any distractions that could possibly skew the data. The data was documented using a Matlab program and hand drawings. The individual attempted to give their full effort throughout the duration of the process, however the authors acknowledge the inevitability of human error and variation in such a test. Nonetheless, the results were relatively consistent and suggest that the metrics presented in this paper accurately predict the time required to reverse engineer tolerances. The actual extraction results along with the time predictions and errors are listed in Tab. 2.

### 5 Concluding Remarks

In this paper, we have presented general metrics for evaluating the time to reverse engineer the tolerances of a product, which is a continuation of the research previously done by Harston and Mattson in [7]. Extraction of tolerance information from a product requires reverse engineering multiple samples of the same product. An exponential decay function adequately describes the relationship between unextracted information remaining in a product and time for the first reverse engineering product sample. With subsequent samples, the relationship becomes more linear, due to changes in the flow of information. We have introduced supporting metrics that characterize the change in information flow rates. The metrics are adapted from the electrical relationships governing the step response of a basic resistor-inductor circuit. A simple example involving 30 ballpoint pens has been offered to both demonstrate the use of the metrics and serve as empirical validation. The example confirms that as reverse engineering samples increase, the flow rates at all unextracted information levels increase toward the same asymptotical limit - the theoretical fastest flow rate, much like the response of electrical current in the resistor-inductor circuit. Moreover, the example suggests that if certain information is known about a product, the person reverse engineering, and the product sample size, then the metrics accurately estimate the total time needed to reverse engineer dimensional tolerances. In the case of the ballpoint pen and cap, the metrics predicted the tolerance extraction time to within 10% error.

Although this paper focuses on dimensional tolerance information, the metrics defined here can also apply to other information types contained by a product. Properties such as electrical conductivity, elasticity, tensile strength, or color may need to be within an acceptable tolerance range as specified by a designer.
in order for a product to fulfil its design purpose. Further study is aimed at validating the metrics for these different information types.

Acknowledgements

This research was partially supported by National Science Foundation Grant CMMI-0800904 for Christopher A. Mattson and Brent L. Adams.

REFERENCES