Measurement and Characterization of Dynamics in Machining Chip Segmentation

E. Whitenton, J. Heigel, R. Ivester
National Institute of Standards and Technology, Gaithersburg, Maryland, 20899-8223 USA
Eric.Whitenton@Nist.Gov

Abstract
Comparing measured chip formation characteristics to those predicted by machining models is an important technique for testing the validity of those models. These characteristics include whether a chip is continuous or segmented, as well as the frequency at which segments form. In this paper, various techniques for measuring segmentation data are reviewed. A sample data set is then used to compare techniques for analyzing segment formation. These techniques show that there is interesting, nonrandom behavior in the sequence of segment formation times. A pattern matching technique is proposed which shows potential to provide valuable additional insight.

1 INTRODUCTION

Studying how metal deforms and flows during the creation of machining chips yields important insights into the metal cutting process. Measurement of the process may be used to improve and verify the accuracy of finite element modeling (**FEM**) simulations. These simulations are an important tool for process optimization, allowing industry to make parts faster, better, and at less cost.

An important measurement is segmentation formation time (**SFT**) the amount of time required for chip segments to form. A segmented chip has alternating zones of low and high shear strain. Figure 1 shows a segmented chip.

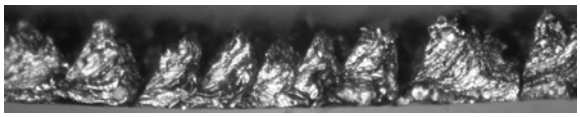

Figure 1: Unetched American Iron and Steel Institute (AISI) 1045 steel machining chip. Width and shape of segments may vary widely. Segments are approximately 0.6 mm high.

This paper investigates methods to measure and analyze SFT data. First, three techniques for measuring SFT are reviewed. Next, a data set of 2 495 segment timings is obtained using one of these methods and analyzed using a variety of techniques. Some of these analyses indicate interesting dynamic phenomena regarding segmentation. Finally, a pattern matching technique is proposed which shows potential to provide valuable insight into the segment formation process. However, this paper focuses on measurement and analysis techniques. A full discussion of all the physical processes which may be involved, such as workpiece damping of tool vibration, is beyond the scope of this paper.

2 MEASUREMENT OF SFT

There are several techniques for measuring SFT. The following sub-sections briefly review and compare three common methods. The first two are well established while the third is relatively new, made practical due to recent improvements in digital high-speed imaging cameras.

2.1 Post-Machining Examination

In post-machining examination, collected chips are mounted, polished, etched, and

examined under a microscope to determine the sizes of the segments. Assumptions are made regarding the chip velocity during machining to convert the measured segment sizes to SFT.

An advantage of this technique is that examination of the chips may yield other important information such as which metallographic phases are present. However, the process is both time and labor intensive. Thus, the sample size (number of segments characterized) is generally small. Since a given population of segments sometimes has a wide range of sizes, a small sample size may result in poor characterization of the average SFT. Also, time-dependency (dynamics) of SFT is difficult to obtain. Finally, individual segments are not always easy to visibly distinguish from each other, even when polished and etched.

2.2 Single Sensor

A second technique used to measure SFT involves recording data from a single sensor during machining, such as acoustic emission or force as a function of time. When a peak detection algorithm is applied to the data, one may determine SFT.

While this technique is relatively easy to implement and yields large data sets, there is some uncertainty as to whether the peaks in the sensor data accurately correlate to segment formation. If other phenomena produce large peaks, or the peak detection algorithm is not set to a correct threshold, results may have large errors with no mechanism for validity checking.

2.3 Image (Array Of Sensors)

A third technique for measuring SFT uses an array of sensors to form a **movie** (a sequence of individual images). Typically, these sensors record visible light, but there is no reason why other characteristics such as emitted thermal radiation could not be used. Image analysis of the movies yield automated SFT determinations [1].

Among the advantages of this technique are that movies may yield additional information such as strain fields, which may be directly compared to FEM simulations. Also, manual viewing of the movies may allow verification of the accuracy of the SFT determination if desired.

Disadvantages include the fact that image processing of such large data sets is computationally intensive. Also, many image analysis algorithms have parameters which must be determined before accurate information is produced. Also, it is often a challenge to acquire clean images without error sources, such as flying chips obscuring the field of view or the images going out of focus due to changes in the chip width.

3 THE DATA SET ANALYZED

Once SFT data has been measured, there are many techniques available to analyze it. Each yields different types of information describing the SFT and how it varies with time. Section 4 compares several of these analysis techniques. A data set, acquired using the acquisition technique outlined in Section 2.3, is analyzed as an illustration. Other publications [1, 2] fully document this data set. For convenience, it is briefly described next.

An orthogonal cut was made on the edge of an American Iron and Steel Institute (**AISI**) 1045 steel disk. The surface speed was 250 m/min and the feed rate was 0.30 mm per revolution. During machining, a high-speed visible light camera acquired a 128 pixel by 128 pixel movie of the side of the chip at 60 000 frames per second. The resulting 21 890 frame movie was then image-processed to determine a set of displacement vectors describing how different portions of the images moved from one frame to the next. The vectors near the cutting zone were then analyzed, and peaks

in their mean orientation were detected to determine when individual segments formed. It was assumed that the side of the chip was a reasonable representation of the entire chip width. Finally, several sections of the movie were manually examined to verify correctness of the SFT data.

Figure 2 shows measured cutting forces as the tool plunges into, cuts, and finally retracts from the workpiece. SFT data includes neither the very beginning nor the very end of the cut because the chips were too small to image with adequate resolution during these times. 2 495 segments were detected and analyzed. The first (approximately) 20 % of the data occurred as the tool plunged into the workpiece, the middle 60 % at nominal steady state cutting, and the final 20 % as the tool was retracting.

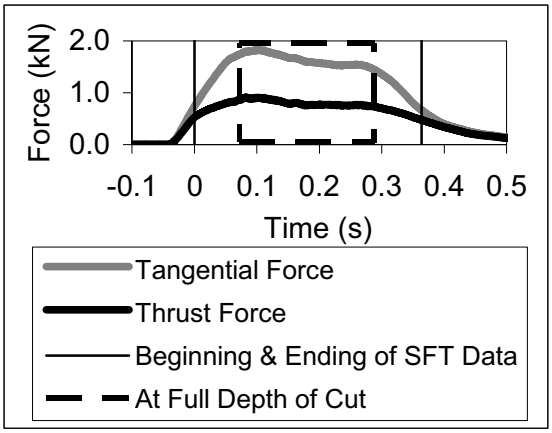

Figure 2: Cutting forces (in kilonewtons) during the test. Time t = 0 is set to the beginning of the SFT data.

4 ANALYSIS

When analyzing the SFT data, there are two issues to explore. First, what is the local average of the SFT values? Second, what are the dynamics of the SFT values? The local average used was the running average of *N* adjacent SFT values, where *N* was generally 50. We refer to the local average of the SFT as **Characteristic 1**. In signal processing terms, Characteristic 1 represents the constant and low frequency components of the SFT data. It reflects both steady state and relatively slowly varying effects such as the tool heating up or the depth of cut slowly changing.

Figure 3 shows both raw SFT data and Characteristic 1 (the gray line) as a function of time. Individual formation times have discrete values because they are computed from the number of movie frames it took for the segment to form. Characteristic 1 changes from about 80 µs at the start and end of the data set to about 165 µs for most of the full depth of cut. Figure 3 also illustrates how individual SFT values fluctuate around Characteristic 1 by as much as 300 %. This fluctuation can be problematic for a researcher using FEM simulations to predict Characteristic 1. Due to long computation times, FEM simulations often encompass only about 10 segments. Even if the FEM model is perfect, 10 segments may not be enough to accurately characterize Characteristic 1 due to the large scatter in the SFT data.

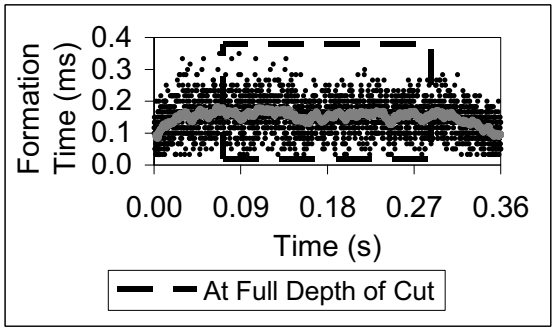

Figure 3: Raw SFT data (black dots) and Characteristic 1 (gray line), the local averages of the formation times, where *N*=50.

The second characteristic of interest is how individual segment formation times deviate from the local average. Referred to as **Characteristic 2**, this characteristic measures both intrinsic noise and higher order dynamics. It is assumed that, with analysis, higher order dynamics information may be separated from noise. In signal processing terms, Characteristic 2 is the high frequency component of the SFT data.

An example of a dynamic effect of interest is if the amount of time it takes a given segment to form directly affects the amount of time it takes the next segment to form. There are several possible physical reasons this might occur. One theoretical scenario might be that a slowly forming segment may tend to leave more of a built-up edge on the tool, or perhaps leave the tool slightly hotter. This might tend to cause the next segment to form slightly faster, which might leave less of a built-up edge or perhaps leave the tool slightly cooler. If this is occurring, a careful analysis of the data should be able to detect the pattern of a slowly forming segment, followed by a rapidly forming segment, followed by a slowly forming segment, and so on. Other physical dynamic mechanisms might produce other patterns. Detection and characterization of these patterns promise to contribute to our ability to understand the machining process.

There are two ways to represent Characteristic 2 data. One is as a **set of deviations**, shown in Figure 4. If a segment formed in 180 μs and the local average is 150 μs, the deviation is +30 μs, or 20 %.

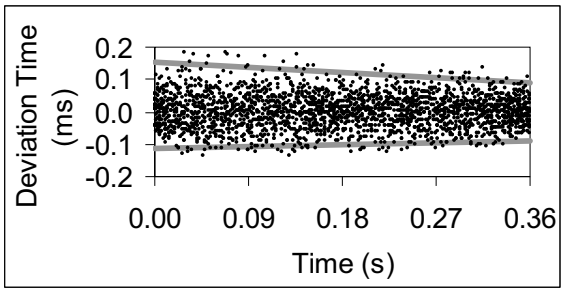

Figure 4: The entire Characteristic 2 data set represented as a set of deviations from the local averages. The gray lines are described in Section 4.1.

Another representation is as a **set of S's and L's**, a portion of which is shown in Figure 5. Segments taking a shorter amount of time to form than the local average are labeled 'S' (represented mathematically as a 0). Those that take longer to form are labeled 'L' (represented as a 1). Some data analysis techniques lend themselves to processing the set of deviations per se while others lend themselves to processing the set of S's and L's.

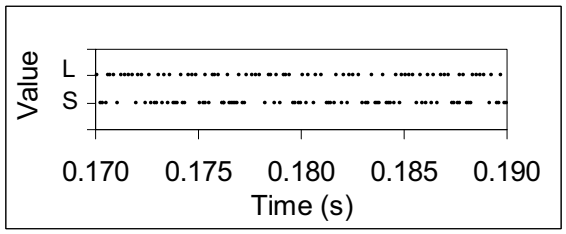

Figure 5: A portion of the Characteristic 2 data represented as a set of S's and L's.

As seen in Figure 6, categorizing whether a segment is an S or an L is not performed solely for convenience of the analysis techniques. Manual examination of the images from which the SFT data was derived has established that when several S's occur in a row, the space between them consistently appears different than when several L's appear in a row. A more careful analysis would be required to establish if there is a consistent pattern concerning the gap between an S and an L.

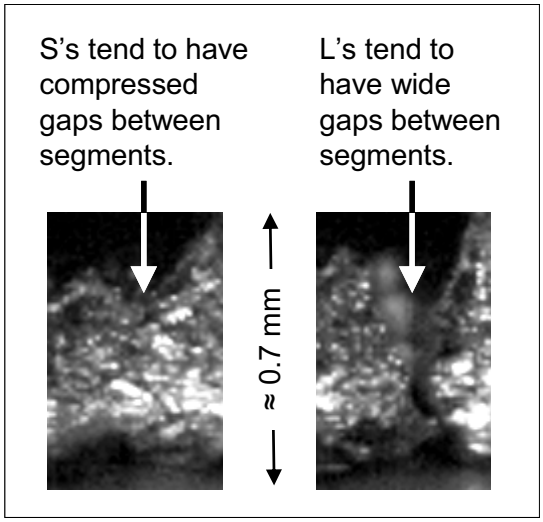

Figure 6: Images from high-speed microvideography of machining chips.

The remainder of this paper is dedicated to analyzing Characteristic 2 data and assessing if different analysis methods yield consistent characterizations of segment formation dynamics.

4.1 Variation Analysis

Levene testing [3] breaks a data set into blocks and determines if the standard deviation of any of the blocks differs from the rest by a statistically significant value. A **block** is a sub-set of the data. The set of deviations was broken into 50 blocks of (approximately) 50 data points. The Levene test showed a statistically significant change in standard deviation with time.

Figure 7 shows how the standard deviation decreases with time. The range of the data (the difference between the maximum and minimum values) also decreases with time, as shown by the two gray lines in Figure 4. The top line was determined by performing a least-squares fit of the maximum value in each of the 50 blocks to a line. The bottom line was determined by fitting the minimums. Note how they come closer together as time passes.

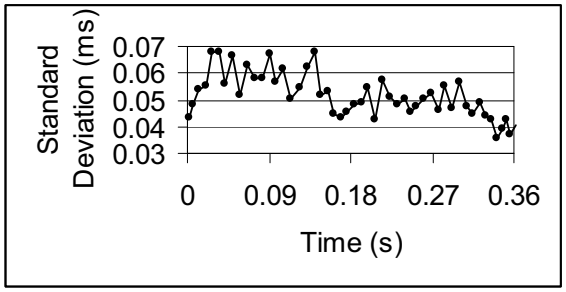

Figure 7: Standard deviation as a function of time for the set of deviations.

Measurements indicate that temperature of both the tool and the workpiece are increasing during this same time. This indicates that temperature may possibly affect variability of the SFT.

4.2 Autocorrelation Testing

Autocorrelation [3] measures the likelihood a change in value at some time will correlate to a change in value at a later time. Autocorrelation has values between -1 and +1. If the autocorrelation is -1, a high value will likely be followed by a low value and a low value will likely be followed by a high value. If the autocorrelation is +1, a high value will likely be followed by another high and a low value will likely be followed by another low. An autocorrelation of 0 indicates that the second value is equally likely to be either higher or lower than the first (no correlation). Units of time are expressed as **lag**. One data point later in time is a lag of 1. For our data, a lag of 1 corresponds to one segment later.

The entire set of deviations, as well as 5 subsets, was analyzed. In all cases, there was a statistically significant positive autocorrelation at lag 1 but none for longer lags.

Portion of Data	Autocorrelation at lag=1
All data points	+0.24 ± 0.04
1st 20 % (plunging)	+0.19 ± 0.09
2nd 20 % (full depth)	+0.28 ± 0.09
3rd 20 % (full depth)	+0.24 ± 0.09
4th 20 % (full depth)	+0.31 ± 0.09
5th 20 % (retraction)	+0.17 ± 0.09

Table 1: Autocorrelation coefficients at lag=1 for subsets of the set of deviations.

Recall from Figure 3 that Characteristic 1 is different while plunging and retracting than it is at full depth. This indicates that a fixed frequency effect, such as a resonance in the structure of the machine tool, is unlikely to be the cause of this autocorrelation.

4.3 Chaotic Dynamics

Previous researchers [4, 5] have explored the use of chaotic dynamics to describe machining data. When chaotic dynamics accurately describes a physical system, it is a powerful tool for understanding a process. However, this technique tends to require very large data sets. Also, issues such as false neighborhoods [4] can be difficult to detect and can adversely impact the results. It is a matter of debate whether our data set is large enough for analysis. Due to these difficulties, we did not pursue this technique.

4.4 Randomness Testing

Randomness testing seeks to determine if a data set is truly random. This is accomplished by applying a battery of tests, each looking for a specific type of nonrandomness. Some tests require a large data set while others do not. If any test indicates nonrandomness, the data is considered nonrandom.

STS [6] is a general-purpose binary randomness testing software package. It was used to perform tests on the set of S's and L's. The data set is nonrandom if any of the following tests indicate nonrandomness by yielding a P value less than 1 % [6]:

1. Proportion of S's and L's for the entire sequence.
2. Proportion of S's and L's for nonoverlapping blocks in the sequence.
3. Maximal excursion from zero of the random walk defined by the cumulative sum of adjusted (-1, +1) digits in the sequence.
4. Is the number of runs of S's and L's of various lengths as expected for a random sequence? A **run** is the number of consecutive occurrences of the same value.
5. Is the number of occurrences of predefined patterns as expected for a random sequence?

These tests were run multiple times with different sequence and block sizes. Test 1 and test 3 consistently passed (did not indicate nonrandomness). Test 2 and test 4 consistently failed (indicating nonrandomness). Test 5 failed for most block sizes. These results show that the SFT data is nonrandom.

4.5 Traditional and Inclusive Runs Analysis

The traditional runs analysis performed by test 4 in Section 4.4 showed that our data is nonrandom. Figure 8 and Figure 9 show results of a traditional runs analysis on the set of deviations performed independently of the analysis in Section 4.4. Run length is the number of consecutive decreases or increases in the data values. The *Y*-axis is the difference between the number of runs found in the data and the number of runs predicted for a random sequence by Monte Carlo simulation. The error bars indicate ±2σ of the random sequence Monte Carlo results. Any value sufficiently far from zero to fall outside of these bars is considered statistically significant. Figure 8 addresses how the number of runs of *exactly* length *i* compare to what is expected for a random sequence. By contrast, Figure 9 examines how the number of runs of *at least* length *i* compare to what is expected.

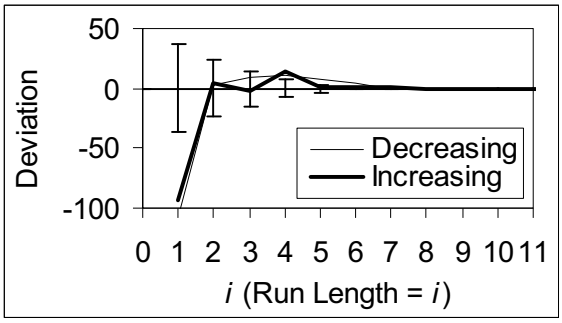

Figure 8: Traditional runs analysis plot for the set of deviations. Run length = *i*.

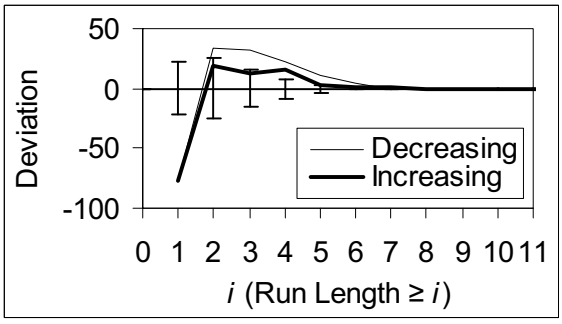

Figure 9: Traditional runs analysis plot for the set of deviations. Run length ≥ *i*.

Figure 10 is similar to Figure 8 except that it is performed on the set of S's and L's. Run length is the number of consecutive S's or L's in a row. Figure 8, Figure 9 and Figure 10 all indicate that, when compared to a random sequence, there are too few short runs (with values below the error bars) and

too many long runs (with values at or above the error bars).

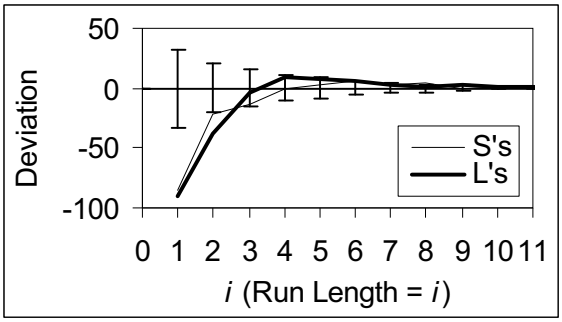

Figure 10: Traditional runs analysis plot for the set of S's and L's. Run length = *i*.

Figure 8 and Figure 10 may be thought of as histograms. Whenever an event is detected, say a run of length 6, the associated bin (bin 6) is incremented by one. However, we argue that a run of length 6 could actually be 2 consecutive runs of length 3. The run of length 6 may have "stolen" counts from bin 3 and put them in bin 6 instead. It is not immediately obvious which interpretation has the best correlation to physical processes involved in machining. Figure 9 addresses this to some extent. A run of length 6 increments bins 1 through 6, each by 1. However, if a run of length 6 is actually 2 runs of length 3, perhaps it should have incremented bin 3 by 2 (instead of 1).

The **inclusive** runs analysis shown in Figure 11 attempts to address this issue. A run of length *i* includes the possibility that it may actually be made up of shorter runs and increments their bins by the appropriate amounts. In Figure 10, bins are incremented only if a run of length *i* is *proven* to exist. In Figure 11, bins are incremented every time evidence is found that there *may* be a run of length *i* hidden in the data. We argue that Figure 10 (where the minimum amount possible was added to the bins) and Figure 11 (where the maximum amount possible was added) effectively determine lower and upper bounds. We propose that the curve best describing machining dynamics lies somewhere between.

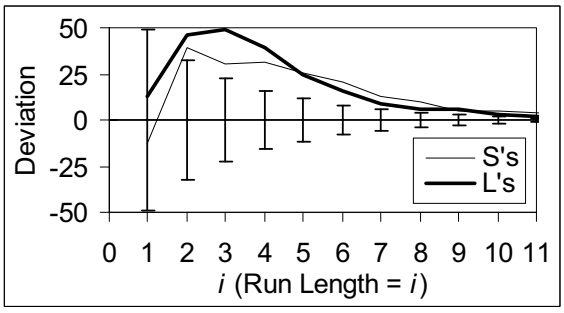

Figure 11: Inclusive runs analysis plot for the set of S's and L's. Run length = *i*.

4.6 Joint Time-Frequency Analysis

Joint Time-Frequency Analysis [7] breaks a data set into blocks and determines which sinusoidal frequencies are present in each block. The output is a surface contour plot where the *X* co-ordinate is time, the *Y* co-ordinate is frequency, and the height of the surface is the amplitude of the sinusoid at that frequency at that moment in time.

One advantage of this methodology is that presenting the data as a surface allows quick visual inspection of how the amplitudes change with time. However, multiple processing settings were tried but little insight was gained. Perhaps modeling the data as a set of sinusoids is not the most useful approach.

4.7 Pattern Analysis

If one views L, LL, … LLLLLLLLLL, S, SS, … and SSSSSSSSSS as patterns, the runs analysis shown in Figure 10 and Figure 11 simply characterize differences between the number of occurrences of those patterns found in the data and the number of occurrences predicted by a random sequence. However, there is no reason to limit the patterns searched for to these. There is also no reason to limit the prediction model to a random sequence. Any pattern and any model of interest may be used. There are multiple ways to implement this analysis. Results for one implementation will be discussed next.

The set of S's and L's was analyzed. The patterns searched for were all possible 1 bit (S and L), 2 bit (SS, SL, LS, LL), 3 bit (SSS, SSL, ... LLL), and 4 bit (SSSS, SSSL, ... LLLL) long patterns. Counting occurrences of patterns was performed in a manner similar to Figure 11. That is, if an SSLL is found, it does not preclude the possibility that the SSLL may also be thought of as an SS followed by an LL. To assess how well various models fit the data, the data was divided into 50 blocks of (approximately) 50 segments. The number of occurrences in each block was then compared to what the model predicted for that block. The average and standard deviation of the results for the 50 blocks was then computed.

Summarized in Table 2, four models were tried and will be described in the text. All four are probabilistic in nature and combine the number of occurrences of shorter bit length patterns to predict the number of occurrences of longer bit length patterns. Due to the way occurrences are counted, there is some inherent covariance between the counts for different patterns. Thus, the probabilistic models are not mathematically rigorous. Nonetheless, it was felt that results from such an analysis would be illuminative. The number of S's is rarely exactly equal to the number of L's for any given block. For example, a typical block might have 47 % of one and 53 % of the other. For our block size, this level of variability is expected. When predicting how many 2 bit long patterns should occur, one might choose to predict 25 % of each. This is what one expects for a random sequence. However, if a particular block has 47 % S's and 53 % L's, predictions of 2 bit long pattern occurrences for that block should take this into account. One expects slightly fewer SS's than LL's in this case. All four models use occurrences of 1 bit long patterns (S and L) to predict those for 2 bit long patterns. Hence, the row labeled 'Predicted 2 bit long patterns are based on' in Table 2 has '1 bit long patterns' in each column. In all four models, the predicted occurrences of the 2 bit long patterns are based on what a random sequence should produce, with a correction for any slight 'imbalance' between the number of S's and L's within the particular block being analyzed.

Average and standard deviation of the errors (across the 50 blocks) for 1 bit long patterns (S and L) are shown in Table 3.

Pattern	Ave. Error	Ave. σ
S	-2 %	2 %
L	+2 %	2 %
SS, LL	+16 %	2 %
SL, LS	-16 %	2 %

Table 3: Average errors and average standard deviations for 1 and 2 bit patterns.

This is well within what one expects for a random sequence. The average and standard deviation of the errors for the 2 bit long patterns are also shown in Table 3.

	Prediction Models:			
	1 Bit	***Previous & 1 Bit***	***Previous Only***	***2 Bit***
Predicted 2 bit long patterns are based on	1 bit long patterns	1 bit long patterns	1 bit long patterns	1 bit long patterns
Predicted 3 bit long patterns are based on	1 bit long patterns	2 bit long patterns and 1 bit long patterns	2 bit long patterns	2 bit long patterns
Predicted 4 bit long patterns are based on	1 bit long patterns	3 bit long patterns and 1 bit long patterns	3 bit long patterns	2 bit long patterns

Table 2: Models for predicting number of occurrences of longer bit length patterns from those of shorter length patterns. Each column describes a model. Each row describes which short bit length pattern occurrences are used to predict longer bit length pattern occurrences.

When rounded to the nearest whole percent, the average and standard deviation for SS and LL were identical, so they are shown together. The same was true for SL and LS. There are consistently more SS's and LL's than expected for a random sequence.

As previously discussed, all four models combine the number of occurrences of shorter bit length patterns to predict the number of occurrences of longer bit length patterns. All four models use 1 bit long pattern occurrences to predict those for 2 bit long patterns. The difference between the models involves how 3 bit and 4 bit long pattern occurrences are predicted.

Model ***1 Bit*** uses occurrences of 1 bit long patterns to predict occurrences of 3 and 4 bit long patterns. This has the effect of predicting what a random sequence should produce, with a correction for any slight "imbalance" between the number of S's and L's within a given block. Model ***Previous & 1 Bit*** combines occurrences for *M*-1 bit long patterns (where *M* is either 3 or 4) with occurrences of 1 bit long patterns to predict occurrences of *M* bit long patterns. Model ***Previous Only*** uses occurrences of *M*-1 bit long patterns to predict occurrences of *M* bit long patterns. Model ***2 Bit*** uses only occurrences of 2 bit long patterns to predict occurrences of *M* bit long patterns.

Looking at errors for the 3 bit and 4 bit long patterns, those which form runs (LLL, SSS, LLLL, and SSSS) generally had the largest errors. There were generally more occurrences than expected. Thus, the results were divided into two groups, one for these patterns (LLL, SSS, LLLL, and SSSS) and one for the other patterns. Shown in Tables 4 and 5, the averages of both the absolute values of the errors and of the standard deviations for each of these two groups were computed as follows: First, the values for the 3 bit long patterns were averaged. Next, the values for the 4 bit long patterns were averaged. Finally, the two averages were averaged.

Model	Ave. |Error|	Ave. σ
1 Bit	48 %	9 %
Previous & 1 Bit	27 %	7 %
Previous Only	19 %	7 %
2 Bit	13 %	6 %

Table 4: Average absolute values of errors and average standard deviations for patterns LLL, SSS, LLLL, and SSSS.

Model	Ave. |Error|	Ave. σ
1 Bit	14 %	5 %
Previous & 1 Bit	10 %	4 %
Previous Only	9 %	4 %
2 Bit	9 %	4 %

Table 5: Average absolute values of errors and average standard deviations for patterns other than LLL, SSS, LLLL, and SSSS.

The model with the lowest average error, combined with a low standard deviation around that error, is considered the best model. Looking at Table 4, model ***2 Bit*** does the best job of predicting occurrences for patterns LLL, SSS, LLLL, and SSSS. Table 5 indicates that both models ***Previous Only*** and ***2 Bit*** are best for the other 3 bit and 4 bit long patterns. Considering Table 4 and Table 5 together, model ***2 Bit*** is the best model. If one considers average errors of about 13 % acceptable, behavior of 3 bit and 4 bit long patterns may be described (or at least approximated) by looking at them as just 2 bit long patterns chained together.

We intend to repeat this analysis for patterns longer than 4 bits in length. It would be significant if model ***2 Bit*** adequately predicts these patterns as well. It would indicate that if you characterize occurrences of 2 bit long patterns, you have characterized occurrences of *all* patterns.

5 CONCLUSIONS

Various analysis techniques have revealed interesting, nonrandom behavior in our machining chip segment formation time data set. Randomness testing showed that the

data is not random. Runs analysis showed a tendency for there to be not enough short runs and too many long runs when compared to a random sequence. Autocorrelation showed a tendency for how long it takes one segment to form to be positively correlated to how long it takes the next to form. In addition, we are developing a pattern analysis technique which shows promise. Preliminary results indicate that the data set might be adequately described as chains of patterns, each pattern being 2 segments long.

Taken together, all these analysis techniques paint a consistent picture; how long it takes for one segment to form is predictive of how long the next will take. It is an open question as to whether this is due to one segment directly affecting the formation of the next, or whether it is due to both segments responding to some external force. However, since the average segment formation time was different while plunging and retracting than it is at the full depth of cut, a fixed frequency effect such as a fixed resonant frequency in the structure of the machine tool is unlikely to be the cause of these results. If a resonance is the cause, the resonant frequency must be changing with the depth of cut.

Finally, variation analysis showed that a drop in variation of the segment formation time data coincided with increasing tool and workpiece temperature. This raises the possibility that these temperatures may affect segment formation time.

This paper focuses on measurement and analysis techniques. A full analysis of physical processes which may be involved is beyond the scope of this paper and left as future work.

6 ACKNOWLEDGEMENTS

The Advanced Manufacturing Systems program at NIST for funding this work. James Filliben of NIST for performing the Levene and autocorrelation analysis, as well as introducing us to runs analysis. Dennis Leber of NIST for introducing us to the STS software. Tim Burns and Jon Pratt, both from NIST, for discussions on chaotic dynamics.

7 REFERENCES

[1] Whitenton, E., Ivester, R., Heigel, J., A Novel Peak Detection Algorithm for Use in the Study of Machining Chip Segmentation, 2007, ISCA 20th International Conference on Computer Applications in Industry and Engineering, San Francisco, CA, pp. 69-76.

[2] Ivester, R., Whitenton, E., Heigel, J., Marushich, T., Arthur, C., Measuring Chip Segmentation by High-Speed Microvideography and Comparison to Finite-Element Modeling Simulations, 2007, proceedings of the 10th CIRP International Workshop on Modeling of Machining Operations.

[3] Filliben, J., DATAPLOT – Introduction and Overview, NBS (National Bureau of Standards) Special Publication 667, 1984.

[4] Abarbanel, H., Analysis of Observed Chaotic Data, 1996, Springer-Verlag New York N.Y., pp. 1-12.

[5] Davies, M., Burns, T., Thermomechanical Oscillations in Material Flow during High-Speed Machining, 2001, Philosophical Transactions: Mathematical, Physical and Engineering Sciences, Vol. 359, No. 1781, Nonlinear Dynamics in Metal Cutting, pp. 821-846.

[6] Rukhin, A., Soto, J., Nechvatal, J., Smid, M., Barker, E., Leigh, S., Levenson, M., Vangel, M., Banks, D., Heckert, A., Dray, J., Vo, S., Pseudorandom Number Generators For Cryptographiic Applications, NIST Special Publication 800-22, January 2001, with revisions dated May 15.

[7] Qian, S., Chen, D., Joint Time-Frequency Analysis, 1996, Prentice-Hall, Englewood Cliffs, N.J.