ARL-TR-9447 • APR 2022

DEVCOM ARMY RESEARCH LABORATORY

Gear Anomaly Detection Using a Matrix Profile Index: Fixed-Axis Gearbox with Cracked Gear: An Application Using Euclidean Distances

by Arnon Hurwitz and Adrian Hood

NOTICES

Disclaimers

The findings in this report are not to be construed as an official Department of the Army position unless so designated by other authorized documents.

Citation of manufacturer's or trade names does not constitute an official endorsement or approval of the use thereof.

Destroy this report when it is no longer needed. Do not return it to the originator.

ARL-TR-9447 ● APR 2022

Gear Anomaly Detection Using a Matrix Profile Index: Fixed-Axis Gearbox with Cracked Gear: An Application Using Euclidean Distances

Arnon Hurwitz and Adrian Hood
DEVCOM Army Research Laboratory

REPORT DOCUMENTATION PAGE	*Form Approved OMB No. 0704-0188*

1. REPORT DATE *(DD-MM-YYYY)*	2. REPORT TYPE	3. DATES COVERED (From - To)
April 2022	Technical Report	April 2019–April 2022

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
Gear Anomaly Detection Using a Matrix Profile Index: Fixed-Axis Gearbox with Cracked Gear: An Application Using Euclidean Distances	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)	5d. PROJECT NUMBER
Arnon Hurwitz and Adrian Hood	5e. TASK NUMBER
	5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)	8. PERFORMING ORGANIZATION REPORT NUMBER
DEVCOM Army Research Laboratory ATTN: FCDD-RLC-IS Aberdeen Proving Ground, MD 21005	ARL-TR-9447

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release: distribution unlimited.

13. SUPPLEMENTARY NOTES

ORCID IDs: A Hurwitz, 0000-0002-6392-6228; A Hood, 0000-0003-4044-0302

14. ABSTRACT

Our purpose is to describe a new anomaly detection method for a cracked spur gear running in a fixed-axis gearbox. Our basic assumption is that the vibration waveform taken from one record of a gear that is running free of anomalies matches best in "Euclidean Distance" to either the record that went before it or the record after. In the present study of six gears, we compare detection outcomes using the new approach with standard vibration-based gearbox Condition Indicator (CI) methods and with deep-learning artificial intelligence (AI) methods exercised on the same data sets. Our method is shown to compare favorably to these alternatives. We describe a new, data-driven anomaly detection method for a cracked spur gear running in a fixed-axis gearbox. Anomaly detection is the first step of a health and usage monitoring system for machinery. The "newness" of the proposed technique resides in the application of a "Matrix Profile Index" to flag anomalous behavior. In the present study, we compare detection outcomes using the MPI approach with standard vibration-based gearbox CI methods and with deep-learning AI methods exercised on the same data sets.

15. SUBJECT TERMS

anomaly, anomaly detection, gear, spur gear, fixed-axis gearbox, gear cracks, matrix profile index, Mechanical Sciences

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Arnon Hurwitz
Unclassified	Unclassified	Unclassified	UU	50	19b. TELEPHONE NUMBER (Include area code) (410) 278-2639

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

Contents

List of Figures

List of Tables

Executive Summary

Our purpose is to describe a new anomaly detection method for a cracked spur gear running in a fixed-axis gearbox. Our basic assumption is that the vibration waveform taken from one record of a gear that is running free of anomalies matches best in "Euclidean Distance (ED)" to either the record that went before it or the record after. In the present study of six gears, we compare detection outcomes using the new approach with standard vibration-based gearbox Condition Indicator (CI) methods and with deep-learning artificial intelligence (AI) methods exercised on the same data sets. Our method is shown to compare favorably to the CI method and quite favorably to the model-intensive AI alternative.

Anomaly detection is the first step of a health and usage monitoring system for machinery. The "newness" of the proposed technique resides in the application of the Matrix Profile Index (MPI)[*†‡] to flag anomalous behavior.

In support of maintaining greater than 90% readiness and providing accurate preemptive fault detection for future vertical lift, existing anomaly detection algorithms will benefit and be supplemented by a model-free anomaly detection method based on a time series construct—the MPI.[*] The result of this investigation compares favorably to a parallel study on the same data that also employs CI to a model-based AI approach.[§] Compared to CI methods, the method described here is more reliable; compared to an AI method the results are equally reliable, require far less data, and is free of parameter-intensive models.

[*] Yeh C-C Mhin-Chia Michael, Zhu Y, Ulanova L, Begum N, Ding Y, Dau HA, Silva DF, Mueen A, Keogh E. Matrix profile I: all pairs similarity joins for time series: a unifying view that includes motifs, discords and shapelets. IEEE ICDM; 2016.

[†] Rakthanmanon T, Campana B, Mueen A, Batista G, Westover B, Zhu Q, Zakaria J, Keogh E. Addressing big data time series: mining trillions of time series subsequences under dynamic time warping. ACM Trans Knowl Discov Sep 2013;Data 7, 3(Article 10):31. DOI:http://dx.doi.org/10.1145/2500489.

[‡] Zhu Yan, Gharghabi S, Silva DF, Dau HA, Yeh C-C M, Senobari NS, Almaslukh A, Kamgar K, Zimmerman Z, Funning G, Mueen A, Keogh E. The Swiss Army knife of time series data mining: ten useful things you can do with the matrix profile and ten lines of code. Data Mining and Knowl Discov. 2020;34(4). ISSN 1384-5810.

[§] Hood A, Valant C, Horney P, Jones A, Lantner JS, Martuscello J, Nenadic N. Autoencoder based anomaly detector for gear tooth bending fatigue cracks. Annual Conference of the Prognostics and Health Management Society; 2021. DOI: https://doi.org/10.36001/phmconf.2021.v13i1.3003.

1. Introduction

The principal prognostic health management (PHM) capabilities for machinery are listed as follows in increasing order of difficulty: 1) Anomaly detection (detects that something is wrong), 2) Diagnostics (infers what is wrong and classifies the failure mode), and 3) Prognostics (projects the damage-in-progress to the point of failure). The lowest level of PHM capability (i.e., anomaly detection) is very important in its own right, and can, over time, be used to attain higher levels of PHM capability.[1] Given our interest in the drivetrains of rotating machinery, we note that drivetrain PHM seeks to interpret signals derived from drivetrain sensors, such as accelerometers, and flag anomalies and/or assign remaining useful devices operating life to the drivetrain.[2] The main interest of this technical report is on anomaly detection only.

A broad 2009 summary of anomaly detection science may be found in Chandola et al.[3] In this work, anomalies are defined as ". . . patterns in data that do not conform to a well-defined notion of normal behavior." The following are concluded: 1) The assumptions drive the success or failure of a method given the situation; 2) Computational complexity is an issue; and 3) Anomaly detection techniques typically assume that anomalies in data are rare when compared to normal instances; however, anomalies are not always rare.

A subsequent 2020 summary[4] compared 88 different Condition Indicator (CI) analyses for gear-crack detection. Its conclusion was that, while several listed detectors had high probability of flagging an anomaly, this "optimal" CI set might change if the anomaly were different than a gear crack.

Methods based on CIs are thus seen to have mixed results; additionally, methods based on artificial intelligence (AI)/machine learning are data intensive and have difficulty converging on a model that generalizes well.[1] Other signal-processing methods often require a high level of expertise to interpret.[5] It would be an advantage to have a reliable, straightforward method that could be adapted to automated real-time drivetrain monitoring for the purpose of general anomaly detection. Existing methods, it will be shown below, are either more variable (the CI approach) or require a sophisticated model-building strategy. The method we describe—the Matrix Profile Index (MPI)[6] using Euclidean Distance (ED) between waveforms—has the potential to be both reliable and to require only sparse parameterization, thus making application less problematic in practice. The algorithm used in our MPI computations, developed by researchers at the University of California, Riverside, is named "STOMP."[7]

In this report we will develop an anomaly detector for a univariate measurement of a fixed-axis gearbox. In addition to being conceptually straightforward, we will show that the algorithm can be configured to use brief samples of data from an ongoing process and thus offer the promise of real-time applications as well as offline fault screening.

We will first describe the concept of the MPI based on an ED metric. We will then describe gear-crack data analysis for six different gears and focus on this data using the MPI for anomaly detection. Good anomaly detection performance is found for our given gear examples. A comparison with two alternative anomaly detection schemes using the same data set is then presented. We end the report with our conclusions and plans for future work.

2. Methods, Assumptions, and Procedures

The MPI is a time-series analysis primitive based on a distance metric. This technology has been under development in recent years by researchers at the University of California, Riverside.[6–11] The technology is readily applicable to many diverse problems ranging from facial profiling to gene sequencing.

The basic inputs to the algorithm are a reference time series (TR) of length n, and a time series query (Q1) of length m (often m<<n), representing a "search span" (SP). In this report, Q1 is taken to be a comparatively short sequence extracted from the comparatively long TR. In other applications it might be a segment taken from a different TR. The parameter, SP, is used to restrict the algorithm search for "best record match" to a contiguous subset of records; we must determine the number of records to include in that search. This number becomes a parameter determined by trial-and-error, but which could be determined by a formal optimization based on the mechanics of a variance comparison that will be described in the following.

2.1 Process

Figure 1 illustrates the process of finding the "best record match" previously mentioned. Let a particular m-point query Q1 (red), be a segment of TR (gray), located at a start position (t1). The Q1 can, in fact, start anywhere along TR (and in the full computation it will start at the far-left end of TR), but for illustration here we pick Q1 as the red-highlighted segment starting at location t1 in Fig. 1.

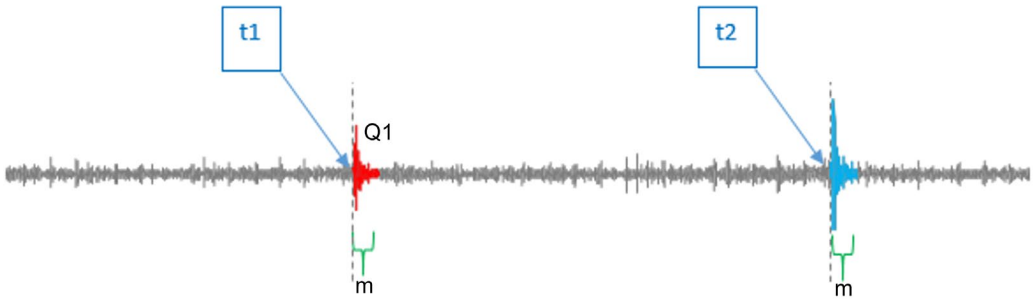


Fig. 1 TR (gray color) with a query Q1 (red color, located at t1) and its closest-distance match (blue color, located at t2); also showing two m-length windows (green brackets) located at t1 and at t2

Figure 1 illustrates how a "short" sequence of data Q1, starting at t1, of length m and highlighted in red taken from a longer sequence TR, highlighted in grey, of length n, is compared against all possible contiguous segments of length m also taken from TR until the location of a "best record match" is found. In this case, the best match is the segment highlighted in blue, is of length m, and starts at t2. The matching is found as follows:

A fixed query, Q1 of length m, starting at the far left of TR, is slid point-by-point along the TR and at each new "window" location, the z-normalized ED between Q1 and the m-length window of TR against which it now stands is computed. The ED is computed as follows:

The z-normalized ED (Appendix A) is given by:

$$d(x,y) = \sqrt{\sum_{i=1}^{m} (\check{x}_i - \check{y}_i)^2}$$

where, $x = \{x_i\}$ and $y = \{y_i\}$ are the query values and windowed values, respectively, $\check{x}_i = (x_i - \mu_x)/\sigma_x$ and $\check{y}_i = (y_i - \mu_y)/\sigma_y$ where μ_x *and* μ_y are the respective means, and σ_x *and* σ_y are the respective standard deviations.

This transform scales x and y into standardized ranges, centered at zero, and then finds the "distance" between them as the square root of the sum of squares of their pointwise vertical deltas. This is illustrated in Fig. 2.

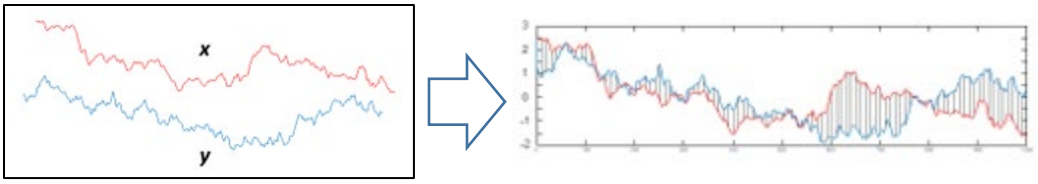


Fig. 2 The z-normalization transformation of two equal-length time series

As we move from the start to the end of this process, we generate a vector of ED distances and a vector of TR locations: one location (a.k.a., index) for each distance. This gives two vectors each of length (n – m + 1). These vectors are each the first row of two (n – m + 1) × (n – m + 1) matrices that we aim to construct based on the distance/location relationships.

Now move the selection of Q1 up along TR by one point to get a "new fixed" version of Q1, and repeat the sliding window computations for a window that, again, travels point-by-point from the first to the $(n - m + 1)^{th}$ location along the TR. A vector of EDs is again computed and this vector becomes the second row of the ED matrix. In parallel, the second row of the location index matrix is also generated.

Continue in this way to move Q1 along TR pointwise and compute matrix rows until the start of Q1 reaches the $(n - m + 1)^{th}$ location of the TR. This produces a final (n – m + 1) × (n – m + 1) matrix of ED distances. Along with the distances, their locations along TR are also collected, and this forms a "location" or "index" matrix of the same dimensions.

The smallest value in each row of the "distance" matrix along with its associated location (i.e., index) value is collected as two new (n – m + 1)-length vectors: the **matrix profile** vector (of distances) and the **MPI** vector (of locations). The MPI records where, along the n-length TR, the closest m-length ED match to any given m-length query occurs. The MPI vector will, as explained in the following, yield the anomaly detector that we seek.

2.2 Exclusion Zone

Not all data points are used in the calculation. To avoid having the location of the shortest distance to a given query be (trivially) zero at or a few points away from that same query, an exclusion zone is imposed that brackets the query (e.g., at t1 in Fig. 1). This exclusion zone is a half-window length (i.e., m/2) on either side of t1. The shortest ED search for the given query occurs across the entire length of the TR (except for the last n – m + 1 points) but excludes this zone.

2.3 Seeded Fault Experimental Data

A detailed description of the experimental setup and data acquisition is given in Hood et al.;[1] the following is a summary. Figure 3 shows a diagram of the experiment. A fatigue crack was introduced into a single tooth of a spur gear and verified. It was then instrumented with a crack propagation sensor and installed inside a gearbox where it was mated with an undamaged gear. The gearbox was operated at a nominal 1500 rpm and 170 ft·lb of torque. Data was captured from an accelerometer mounted to the gearbox housing. Data was sampled at 100 kHz in one-second bursts. A file was saved for each burst. This amounts to approximately 4000 samples for each shaft cycle.

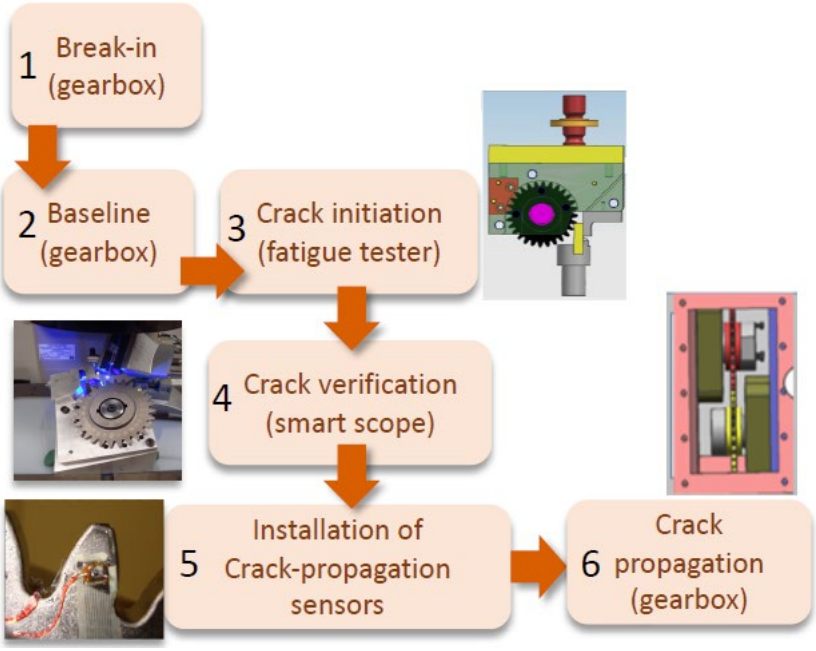


Fig. 3 Gear tooth seeded fault experimental process

Since the rotation speed varies slightly, it is possible that different shaft rotations may have different numbers of points. To avoid this problem in processing, time-synchronous averaging (TSA) is used to convert the data from the uniform time spacing to a uniform rotation angle spacing. This is accomplished by up-sampling the points to a predetermined m-point, where m is usually slightly larger than the average points per shaft cycle in the data. The value chosen is m = 4096.* In this process an accelerometer vector with a length of 100,000 observations, associated

* Because the original number of points hovered around 4000 points, 4096 was chosen as the up-sample value because it is customary in signal processing to use a power of 2 to speed up calculations.

with 1-s of operation, was compressed into the TSA vector with a length of 4096 points.[1] Each TSA vector is referred to as a "record." We chose m = 4096 for this study because it gave satisfactory results and as one record it can be visualized as one rotation's worth of (averaged) data; however, other values are possible, such as "half-records," and so forth. We leave such investigations to further research.

For example, a test gear pair that was free of anomalies—209 and its mate 208 (i.e., 208/209)—was run for 2 h, producing 7200 files. Each file was processed using TSA and stored as a record. This collection of data is called the "baseline" for gear 209. A crack was then initiated on a spur tooth of the same gear and the gear was run on the same bench setup until the tooth broke. This occurred after about 4 h, producing about 15,067 records. This collection of data was termed the propagation data for gear 209.

To expand our investigation beyond just one gear, similar test runs were performed on several other test gears to introduce variation in the training set and to assess the anomaly detector's performance across different data sets. Five of these additional gears are numbered: 207 (206/207), 213 (212/213), 215 (214/215), 217 (216/217), and 219 (218/219). Numbers of baseline records and of propagation records per gear are given in Table 1.

Table 1 Number of baseline and propagation records per gear

Gear	207	209	213	215	217	219
Baseline records	7200	7200	7200	7200	7200	7200
Propagation records	5477	15,067	17,176	12,771	8600	9306

For this work, the concatenated collection of TSA records for either the baseline case or the propagation case is the TR. For example, for gear 207 and baseline data, there will be 7200 × 4096 = 29,491,200 points in the TR. In addition, each record will serve as an m-length query.

2.4 MPI: Details for a Rotating Gear

We introduced the construction of the matrix profile and MPI in Section 2.3. Henceforth we will consider only the MPI because it has a direct application to gearbox anomaly detection.

Regarding non-anomalous gear behavior, or stability, we have assigned m = 4096 as the number of TR points counted in a single record of the gear under investigation. In addition, we assume that a baseline TR is a TR that embodies

stability with respect to the rotating gear. In this case, we hypothesize that the location of the shortest ED to any given query within that TR that contains one record's worth of m-points is likely to lie in the window that contains the next (or previous—by symmetry) record's worth of points.

One might wish to extend this definition of stability across the entire length of the TR, but there are two reasons why this is not advisable: 1) it is computationally burdensome because we are dealing with millions of points and the computational burden is of $O(n^2)$;[7] and 2) a narrow locality of a query is sufficient for our purposes.

2.5 Procedures Using ED

For purposes of exposition, let us analyze the baseline data from gear pair 209/210. Figure 4 shows an MPI plot in which the MPI is plotted against the query's beginning index value. A diagonal is plotted with a slope of 1 to highlight the current Q1 start position. For example, if the m-length query starts (arbitrarily) at index = 1000, the window of m values whose ED is smallest has a starting index = 5096, indicating 4096 points above the diagonal. Another query starting at index = 16,384, the window of m-values whose ED is smallest has a starting index = 12,288. This indicates 4096 points below the diagonal. The black points form parallel "railroad tracks" on either side of the blue 45° line. One can see how this plot can be used to locate points, or a collection of points whose closest record is not immediate. The takeaway from these results is that, for the 10 records pictured, the gearbox readings are stable as the MPI selects minimum ED locations from exactly the next or previous record of the gear. This is intuitively reasonable. If the minimum EDs were at random, unpredictable locations would suggest unstable gear behavior as we might expect a crack in a gear to generate mechanical "interference" in the accelerometer readings.

Figure 4 is a snapshot that only shows 10 records for illustration. Recall, the baseline case has 7200 records, and the propagation cases contain considerably more. To avoid clutter, select snapshots along the TR are plotted; these were selected based on our computational resources. In practice, the number of snapshots would have to be determined according to the needs of the detector to pick up anomalies according to the needs of the application.

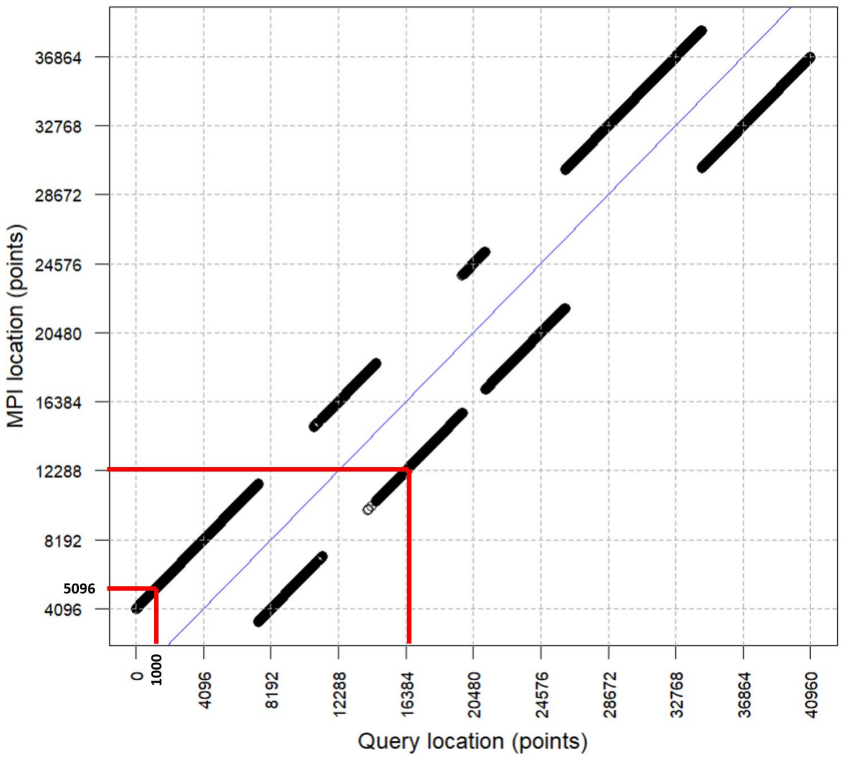


Fig. 4 MPI plot – Gear 209 baseline: 10 records of 4096 points each with a 45° reference diagonal shown in blue ("location" refers to point location)

A typical result is shown in Fig. 5 for the propagation data in which 12 snapshots are used. Now, the x-axis tick marks represent the record number and the MPI values are plotted between tick marks. Some values are off the main two sub-diagonals. These would suggest unstable gear behavior. The 12 snapshots represent $(10 \times 4096 \times 12/61714432) \approx 0.8\%$ of the data. When examining the MPI plots for these propagation snapshots, for the most part the location for a given record (of length 4096 points) is no longer one record distant but is mostly scattered over the many possible locations within the snapshot.

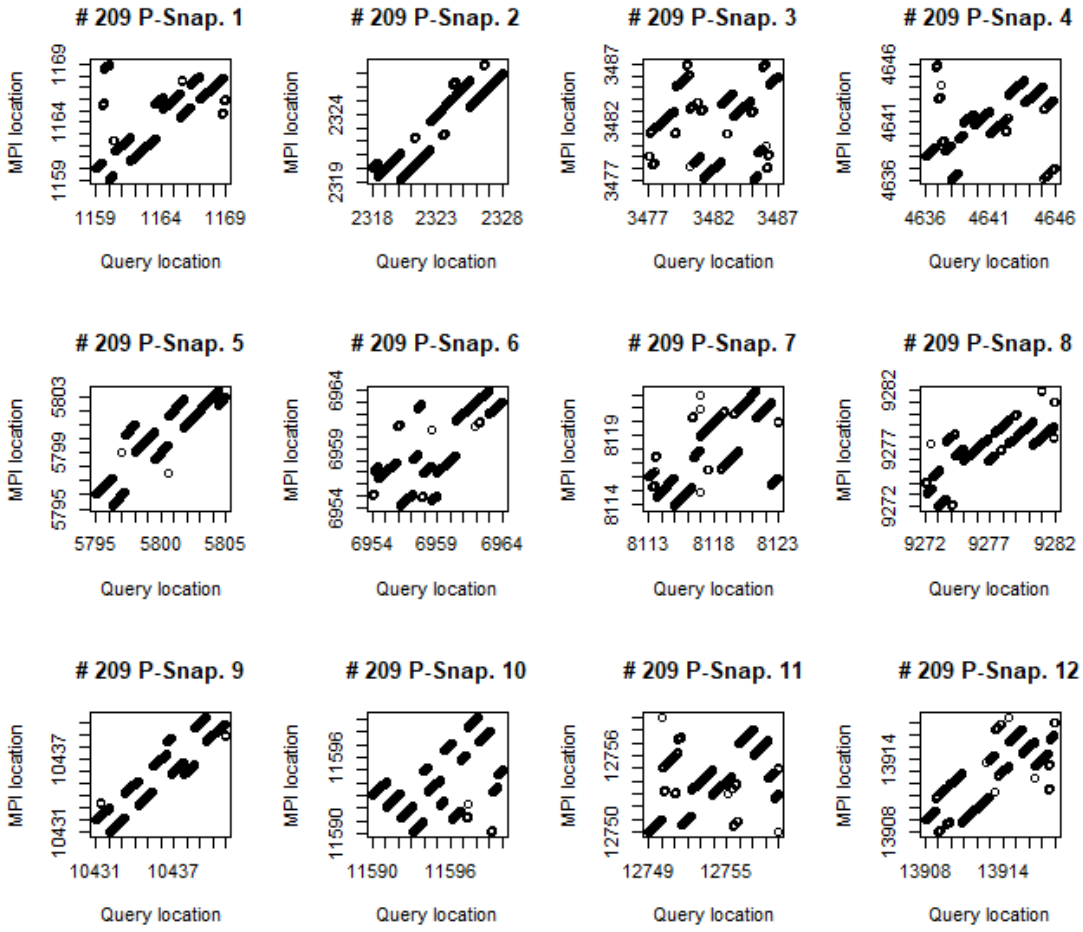


Fig. 5 Gear 209 propagation data – MPI snapshot plots 1–12 ("location" refers to record location)

Figure 6 shows the same type of 12 snapshots for gear 209's baseline data. Here, the first six snapshots appear quite "regular" in the sense of Fig. 4, with parallel off-diagonal location lines, whereas the last six snapshots of Fig. 6 indicate a breakup of such regularity.

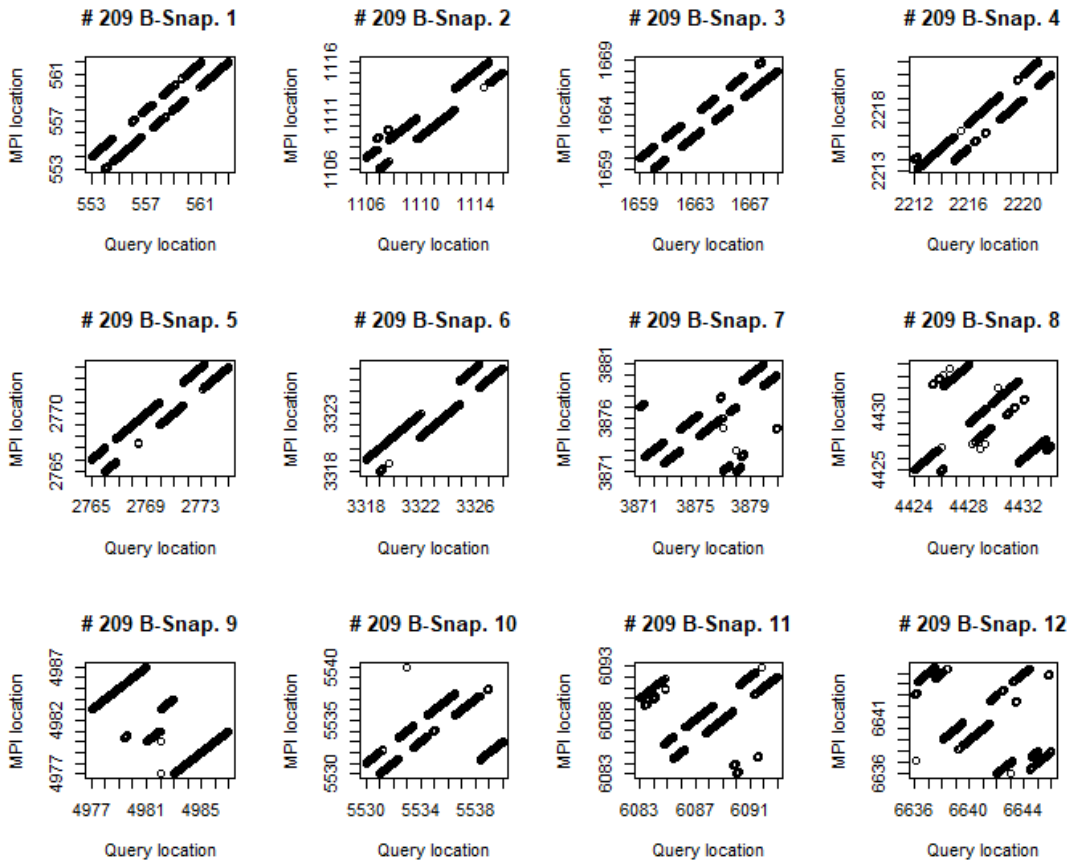


Fig. 6 Gear 209 baseline data – MPI snapshot plots 1–12 ("location" refers to record location)

2.6 Variance of MPIs Within a Snapshot

The variability of the MPI for propagation data can be quantified within a snapshot by computing the variance of the absolute vertical deviations of each MPI value from a 45° reference diagonal. Each snapshot will give one value of variance. We expect—if our hypothesis of anomalous behavior holds true—that a "baseline" MPI location plot would have variances for all snapshots equal to zero, and "propagation" MPI location plots will have variances greater than zero. If this were indeed true, we could infer that a significantly higher average variance for the propagation series over that for the baseline will be an indicator of an anomaly.

It should be noted here that the actual values of the variance sequence are not of concern; it is the overall relative size of the propagation variances versus that of the baseline variances, for a given gear, that is of interest. The term "significantly higher" in the previous paragraph is used in the statistical sense—actual values are not of interest, only "significantly different" values apply to our argument here.

This method would, at first, appear to be a good approach to flagging anomalous behavior for a cracked gear, and this would be true if a similar analysis of baseline gear data gives a trace of all or many zero variances across all snapshots of such data for a given gear. However, this does not hold across all snapshots for the baseline data of gear 209. This can be seen in Fig. 7, which shows the variance versus snapshot number for both the baseline and propagation cases.

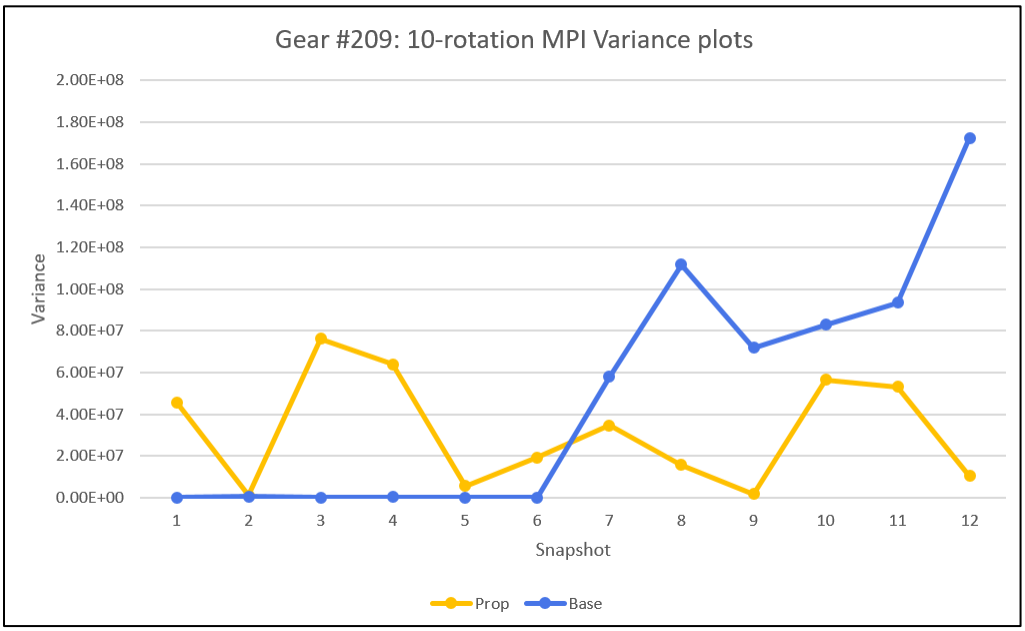


Fig. 7 Gear 209, 10-record variance snapshots for propagation and baseline MPIs*

Only half of the baseline data snapshot variances are zero or close to zero.

As can be seen from Fig. 7, our desired relative variance size differential between baseline and propagation data did not supply a convincing anomaly signal based on a 10-record analysis. We conjecture that a 10-record snapshot may allow too much latitude to the matrix profile method to locate a closest-matching record of 4096 points to a given record—and therefore produce the random scatter seen across several panels (i.e., the last six snapshots) of Fig. 6. We also observe this in the variance plot. Thus, we look at a narrower search span for both propagation and baseline data—that of five records—to see if a reliable anomaly detection signal might be derived. This five-record choice was determined by a trial-and-error method but could be found using an optimization approach for different anomaly/gearbox types.

* There is no relationship between the snapshot index number of baseline and that of propagation. It should be interpreted as ordered categories of equally spaced spans from start to finish of either baseline or propagation test.

2.7 Five-Record Snapshot MPI Analysis

Examining the 12 plots (snapshots) of Fig. 8, the x- and y-axis ranges cover the lower-left quadrant of those in Fig. 7—that is, half the number of records. The MPI locations are, however, similarly scattered compared to those of Fig. 5. This is expected, given our conjecture about the MPI in the presence of an anomaly. The variance plot for Fig. 7 is shown in Fig. 10.

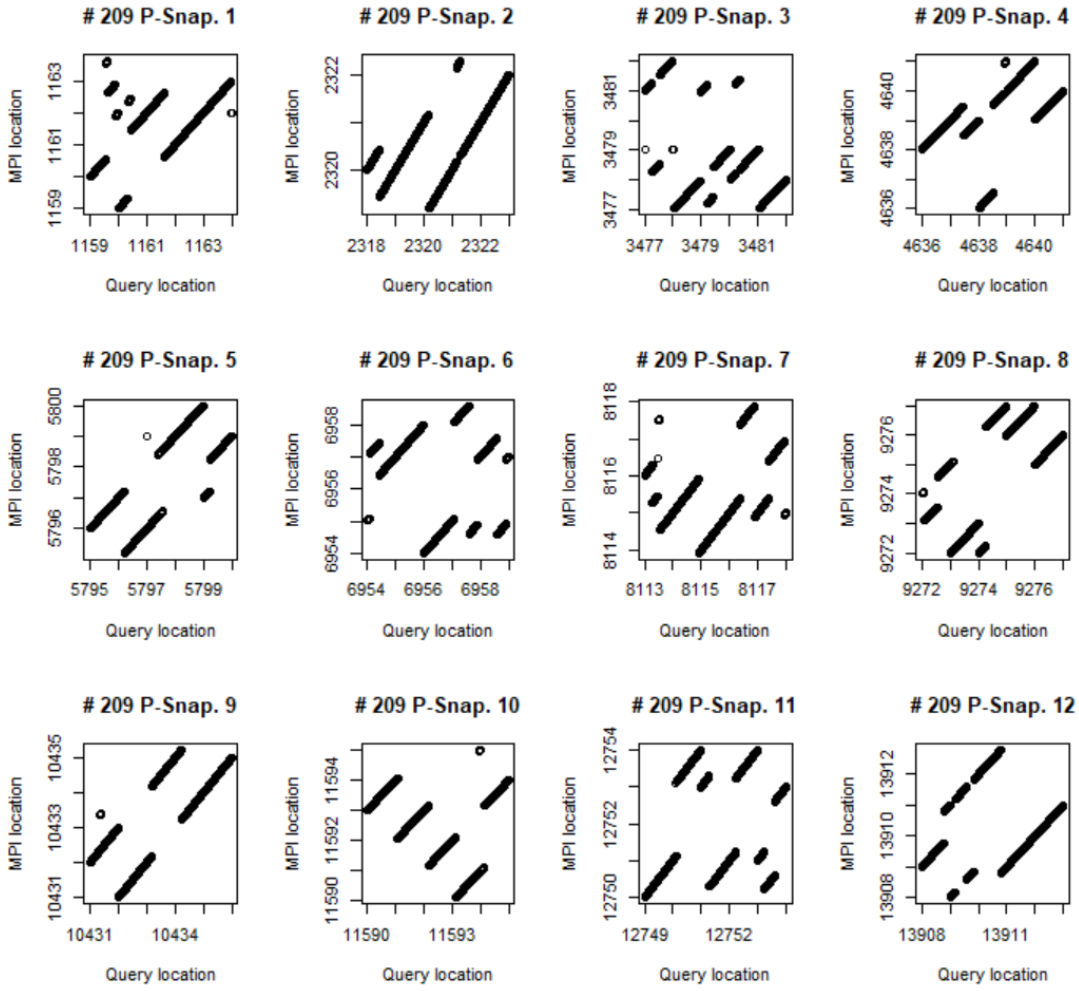


Fig. 8 Gear 209 propagation data – MPI plots of 12 five-record snapshots (45° line not shown; "location" refers to record location)

Figure 9 shows the 5-record baseline results. Here we see a "better" outcome in the sense that restricting the number of records has worked to stabilize the plots so that baseline data produces the expected "parallel lines" outcomes. This suggests that, within the space of any five sequential records selected from the TR, a given record most closely matches in ED with either the previous or the subsequent record—an indicator of "stability," or non-anomalous behavior.

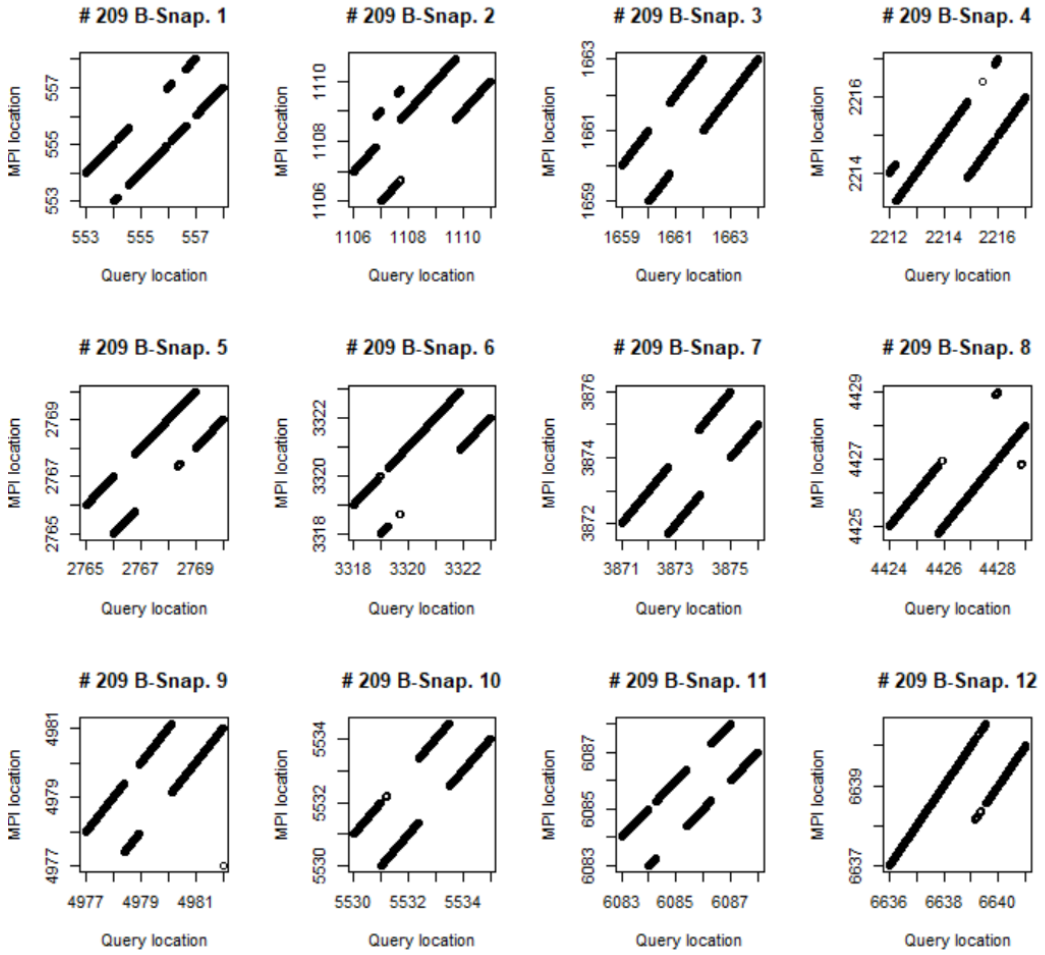


Fig. 9 Gear 209 baseline data – data snapshots 1–12: MPI plots

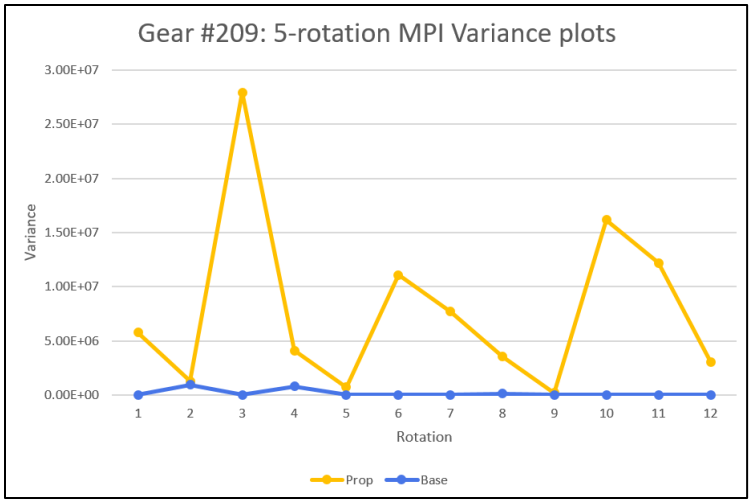


Fig. 10 Gear 209 five-record variance plots for propagation and baseline MPIs

3. Results

Figure 9 shows that by restricting the MPI algorithm's ability to search for Euclidean closest-matched distances within a 5-record range for the baseline data gives a far more regular result than for the 10-record case. It suggests that we have the ability to detect, graphically, the presence or absence of a gear-crack anomaly (or anomalies) by using the MPI variance plots, over five-record snapshots. This result applies to gear 209 data. It is now of interest to see if this conclusion applies equally to the other gears of the gear-crack data set.*

Appendix B gives the 12-snapshot plots for propagation and baseline data for all six gears.

Figure 11 gives the variance snapshot plots, derived from the results graphed in Appendix B, for all six gears.

* The snapshots taken over the length of the TR will necessarily be a judicious choice given the length of the time series as well as the computation speed available.

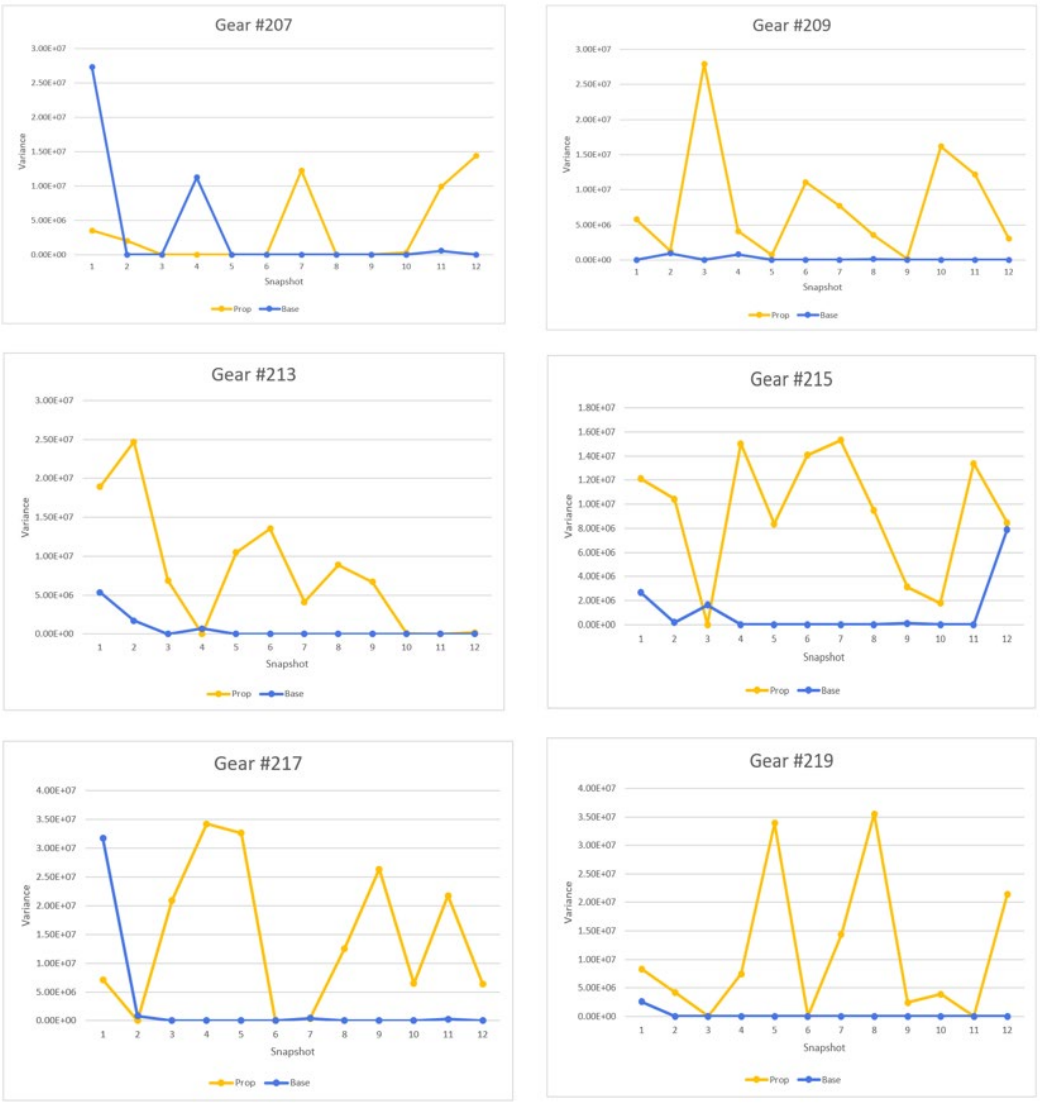


Fig. 11 Five-record variance plots for propagation (orange) and baseline (blue) MPIs

3.1 Hypotheses and Tests: Do Propagation Variances Show Significantly Different Behavior?

Figure 9 suggests that baseline "snapshot" variances, apart from a few high values, equal zero, whereas propagation variances are, for the most part, greater than zero. The variance data for all 72 baseline and 72 propagation snapshots are assembled in Appendix C and show that, here, there are 12 zero-variances for propagation and 51 for baseline data. Taking the baseline variance behavior as a standard for indicating a non-anomalous situation, we are interested in whether the propagation variances show significantly different behavior.

Figure 12 gives boxplots for propagation and baseline variances. Overall means that the collections in each plot are given as horizontal red lines; the mean of the propagation variances is 8.92 e + 06, whereas the mean for the baseline variances is 1.35 e + 06.

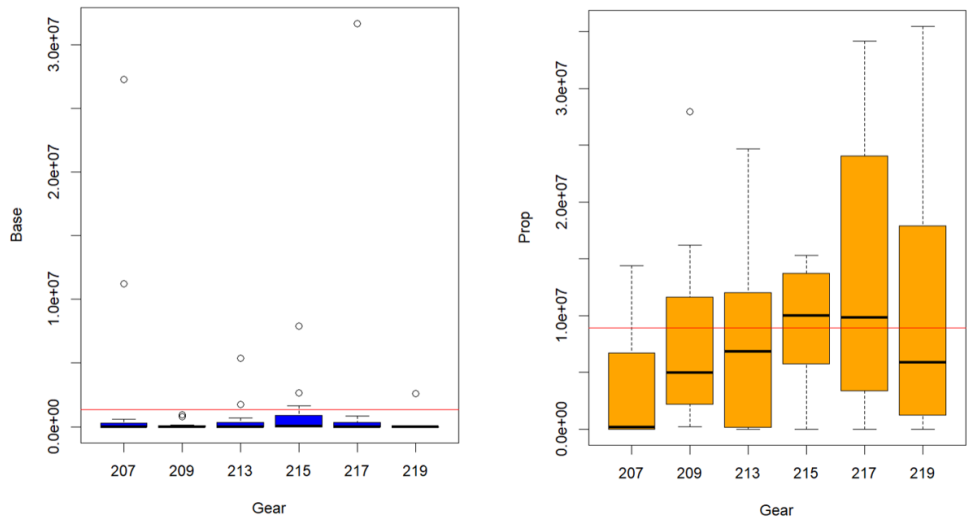


Fig. 12 Boxplots of the baseline and propagation and variance distributions for each gear; (12 variance values per gear). Red lines show overall mean values.

It appears that the propagation variances run much higher than those for the baseline data. Formalizing this statement as a statistical test will give a basis for an anomaly detector. We proceed as follows:

1. Recall that, for each gear, the time-series data (propagation or baseline) was segmented into 12 parts and, for each part, a small number (i.e., 5) of sequential records were selected to compute a single variance. This gave 12 variance values per gear for its own baseline and another 12 for its propagation series. As the 12 variances—either within baseline or propagation sets—were far apart in terms of observations we will regard these 12 variances as random drawings of a variable, that variable being the variance for a snapshot of 5 records.

2. We take the mean baseline variance as a constant and representing a non-anomalous performance standard. Then we test to see if the mean propagation variance—for each gear—is significantly greater than the baseline's. Specifically, we test to see if there is a small probability, P^*, that the average propagation variance is smaller than the average baseline variance, where:

$$\mathbf{P}^* = \mathbf{P}\,(\text{Prop.var.average} \leq \text{Base.var.average})$$

is small. If $\mathbf{P}^*$ is indeed small, we reject the hypothesis contained in the parentheses and conclude that we have an anomalous situation.

3. If we can show that the variance set for a given gear's propagation data is
    a. normally distributed and also
    b. uncorrelated, then we can consider the 12 variances in a given propagation gear's data to be a normally and independently distributed random variable.
4. If 3a and 3b hold, we can perform a hypothesis test on the mean of the set of 12 propagation variances for a given gear using a student's t-test and decide if that mean is different to the overall baseline variance mean.
5. If it is concluded, from the tests in Step 4, that the mean of a set of 12 baseline variances for any given gear is significantly greater than the mean for the baseline, we have a statistical basis for an anomaly detector.

Starting with Step 2:

2. The average overall variance for the $6 \times 12 = 72$ baseline variances is 1.35e + 06. We take this as a constant for test purposes.

3a. Normality test: Bonferroni-corrected Shapiro-Wilk tests for the propagation variances within each gear gave p-values: {0.006, 0.198, 0.708, 1.000, 0.762, 0.072} for gears {207, 209, 213, 215, 217, 219}, respectively. This suggests that, within each gear, the variances are normally distributed (except possibly for gear 207).

3b. Correlation test: The partial autocorrelation functions for any of the six propagation gear data variance series showed no evidence of autocorrelation at the 95% confidence level—all autocorrelations lie within the confidence bounds (horizontal blue dotted lines) as shown in Fig. 13.

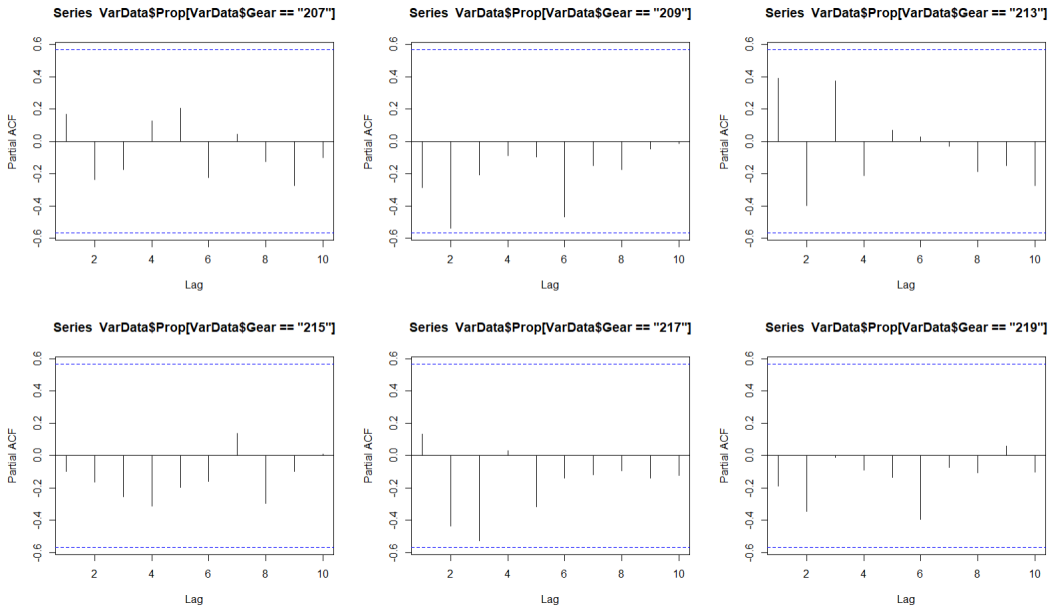


Fig. 13 Partial autocorrelation plots for the six propagation-data gear variance sets

4. Bonferroni-corrected student's t-tests for the six gears, with null hypothesis "Average variance less than or equal to 1.35e + 06" versus the alternative hypothesis of "greater than 1.35e + 06" gives the following probability values, respectively, for the gears as:

Gear number:	{207, 209, 213, 215, 217, 219}
Probability:	{0.57, 0.054, 0.048, 0.000, 0.018, 0.072}.

5. Apart from gear 207, all tests reject the null hypothesis at a better than 90% significance level. We conclude that for all gears—except 207—this test detected anomalous behavior following a crack imposed on a spur gear.

This simple test formalizes what we infer visually from Figs. 9 and 10: runs with a cracked gear produce significantly higher nonzero five-record MPI snapshot variances than non-cracked gears. This idea forms the basis of an anomaly detector. An anomaly needs to be detected before a gear actually fails.

3.2 Comparing the MPI to CI and AI Methods for Anomaly Detection

Considering that the present investigation is based on data arising from measurements taken off a simple, fixed-axis gearbox running in stationary mode under laboratory conditions, it does not seem surprising that a straightforward anomaly detection scheme such as we have demonstrated here should, indeed, discriminate effectively between normal gearbox operation and one with anomalies

present. However, as noted, gearbox anomaly detection (for small damage) has historically been elusive even for the fixed-axis case. These previous studies, mentioned in the Introduction above, were not done using our present gear-crack data, so we now consider a parallel study[1*] on the same data set using standard CIs as well as an "intelligent" AI learning, model-based method based on a neural net (NNET).

3.3 Anomaly Detection by MPI

We will compare the results we found using our MPI method based on the ED metric to those of Hood et al.[1] for the six gear-crack data sets available to us.

Table 2 shows that, with the described anomaly alarm scheme described, anomalies were correctly "red-flagged" in five of six of the gear runs. For this example, this is a 5/6 or 83% success rate for the detector.

Table 2 Anomaly detection via MPI method with ED using 12 snapshots.

Test Gear no.	P* (probability)	Detection decision	Comments
207	0.57	No anomaly	. . .
209	0.05	Anomaly	95% confidence
213	0.00	Anomaly	99% confidence
217	0.02	Anomaly	98% confidence
219	0.07	Anomaly	93% confidence

3.4 Anomaly Detection by CI

Turning to anomaly detection by CI as reported in Hood et al.,[1] we reproduce the graphic found therein as Fig. 14. This graphic depicts the undamaged (blue) versus the damaged (orange) CI distributions. (Note: There are two more gears here than in our previous discussion).

* In the Hood et al.[1] presentation, two additional gears (211 and 221) were exercised over baseline and propagation trials.

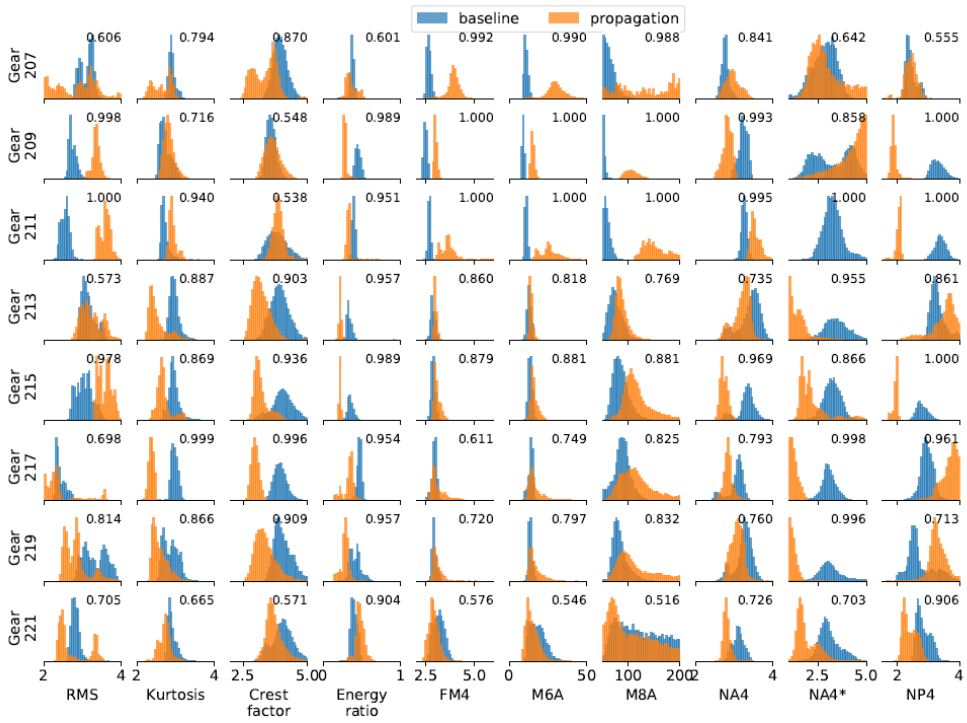


Fig. 14 Anomaly detection by eight condition indicators

There are eight CIs listed along the bottom of Fig. 14 and gear numbers at the left side. Completely nonoverlapping distributions (blue vs. orange) indicate successful anomaly detection.

Of the gears that we had data for, these red boxes include 209 and 217. It appears that the CIs did not generalize well to all the gears: no one CI worked best on each test. Table 3 summarizes data from Fig. 14, where the Area Under the Curve (AUC) is used for the single valued performance metric.[1,12]

Table 3 Anomaly detection in propagation data by CI method

Test gear no.	Highest AUC	Comments
207	0.99	M6A
209	1.0	RMS, FM4, M6A4, M8A, NP4
213	0.96	ER
215	1.0	NP4
217	0.99	Kurtosis
219	0.996	NA4

3.5 Anomaly Detection by AI

Figure 15 displays anomaly detection on the same data set (plus two additional gears: 211 and 221) using AI methods detailed in Hood et al.[1] and using a mean square error (MSE) of baseline versus propagation model error as a goodness-of-performance parameter. Propagation MSE is seen to be higher on average and have in general a much wider spread than baseline MSE. Moving left-to-right across the columns of Fig. 15 we see more (baseline) gear data information added gear-by-gear to the AI estimation algorithm.

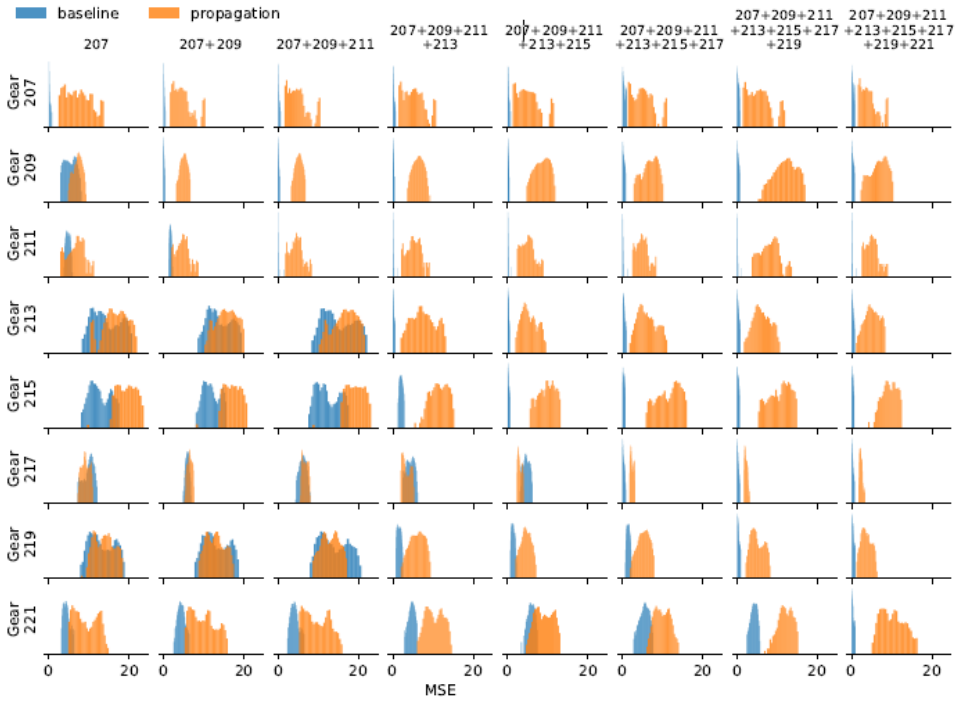


Fig. 15 Anomaly detection by autoencoder AI results

In Fig. 15, autoencoders (one per column) trained on one or more baselines (as indicated on the top of the columns), evaluated over eight gear baseline and crack-propagation tests. Log scales were used in the y-axes to better show the distribution tails.

Going across from left-to-right (Fig. 15) for a given test gear, the general trend is that performance improves as more baseline cases from other test gears are used in the training. However, going down each column (Fig. 15), performance decreases for the gears whose baseline data was excluded from the training. This would suggest that the models do not generalize well from one gear to the next. Further, a model trained on an individual gear and evaluated on that same gear had a wider separation between error distributions associated with baseline and crack-propagation data, as shown in the top-left histogram (Fig. 15, model 207 evaluated

on gear 207). The final column (Fig. 15) contained results for all baseline cases that were used in the training set; AUC values of 1 were obtained for each gear. It is undetermined whether the training set needs to include evaluated gear or if there is some minimum number of test gears needed before the model generalizes well.

Autoencoder-based, data-driven models showed improved and more consistent performance than the classical CIs for the first level of PHM capability (i.e., anomaly detection). To obtain more reliable performance and reduce Type II errors (i.e., false alarms), the autoencoder should be trained on multiple runs of the asset across all different operating and environmental conditions of interest.

4. Conclusions

Our aim was to describe a new data-driven anomaly detection method, the MPI, for cracked gears running in a fixed-axis gearbox. Compared to standard CI methods, it proved more reliable, and was generalizable across the whole data set with a high success rate across six gears. Compared to an autoencoder-based AI method (i.e., autoencoder NNET) it was similarly reliable, used far less data, was model-free, and is readily explainable.

The MPI method requires only limited data and there are no model over-fit or under-fit issues. By using small segments of streaming data, parallel processors, and efficient algorithms, MPI could be sped up to work in real time.[6] MPI methods can be extended directly to other types of vibration profiles.

We consider these findings a notable step forward toward a reliable and straightforward framework for gear anomaly detection.

5. Future Work

The data we analyzed was derived from a laboratory-bench device containing two gears on fixed axes. This simple setup gave sufficient interest for our analyses. Future work will need to be directed to more complex machinery.

6. References

1. Hood A, Valant C, Horney P, Jones A, Lantner JS, Martuscello J, Nenadic N. Autoencoder based anomaly detector for gear tooth bending fatigue cracks. Annual Conference of the Prognostics and Health Management Society; 2021. DOI: https://doi.org/10.36001/phmconf.2021.v13i1.3003.

2. Huang B, Yuan D, Chao J, Lee J. Review of data-driven prognostics and health management techniques: lessons learned from PHM data challenge competitions. Semantic Scholar: Corpus ID: 204956929. 2017.

3. Chandola V, Banerjee A, Kumar V. Anomaly detection: a survey. ACM Comput. Surv. 41. July 2009;3(Article 15):58. DOI: 10.1145/ 1541880.1541882.

4. Bechoefer E, Butterworth BA. Comprehensive analysis of the performance of gear fault detection algorithms. 2020.

5. Lei Y, Lin J, Zuo MJ, He Z. Condition monitoring and fault diagnosis of planetary gearboxes: a review. Elsevier Ltd Measurement. 2014;48:292–305. dx.doi.org/10.1016/j.measurement.2013.11.012.

6. Rakthanmanon T, Campana B, Mueen A, Batista G, Westover B, Zhu Q, Zakaria J, Keogh E. Addressing big data time series: mining trillions of time series subsequences under dynamic time warping. ACM Trans. Knowl Discov 2013 Sep;Data 7, 3(Article 10):31. DOI:http://dx.doi.org/10.1145/ 2500489.

7. Zhu Y, Zimmerman Z, Senobari NS, Yeh C-C M, Funning G, Mueen A, Brisk P, Keogh E. Matrix profile II: exploiting a novel algorithm and GPUs to break the one hundred million barrier for time series motifs and joins. IEEE ICDM. 2016;739–748.

8. Yeh C-C M, Zhu Y, Ulanova L, Begum N, Ding Y, Dau HA, Silva DF, Mueen A, Keogh E. Matrix profile I: all pairs similarity joins for time series: a unifying view that includes motifs, discords and shapelets. IEEE ICDM; 2016.

9. Zhu Y, Gharghabi S, Silva DF, Dau HA, Yeh C-C M, Senobari NS, Almaslukh A, Kamgar K, Zimmerman Z, Funning G, Mueen A, Keogh E. The Swiss Army knife of time series data mining: ten useful things you can do with the matrix profile and ten lines of code. Data Mining and Knowledge Discovery. 2020;34(4). ISSN 1384-5810.

10. Mueen A, Keogh EJ. Extracting optimal performance from dynamic time warping. KDD. 2129–2130. (22nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining). 2016, Aug 13–17.

11. Rakthanmanon T, Campana B, Mueen A, Zhu Q, Zakaria J, Keogh E, Batista G, Westover B. UCR suite for time series subsequence search. 2019. https://www.cs.ucr.edu/~eamonn/UCRsuite.html.

12. Fan, J., Upadhye, S., & Worster, A. (2006). Understanding receiver operating characteristic (roc) curves. Canadian Journal of Emergency Medicine, 8(1), 19–20.

Appendix A. Euclidean Distance Metric*

* Used with permission from Zhu Y, Imamura M, Nikovski D, Keogh E. Matrix profile VII time series chains: a new time series data primitive. U Cal Riverside – tutorial presentation; 2016.

Euclidean Distance Metric

Given two time series

$\mathbf{x} = x_1 \ldots x_n$

and

$\mathbf{y} = y_1 \ldots y_n$

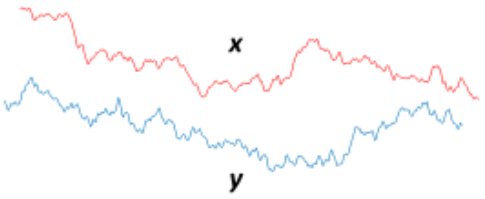
their z-Normalized Euclidean distance is defined as:

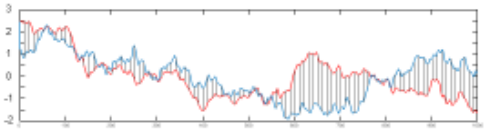
$$\hat{x}_i = \frac{x_i - \mu_x}{\sigma_x} \qquad \hat{y}_i = \frac{y_i - \mu_y}{\sigma_y}$$

```
function y = zNorm(x)
y = (x-mean(x))/std(x,1);
```

$$d(x,y) = \sqrt{\sum_{i=1}^{n} (\hat{x}_i - \hat{y}_i)^2}$$

```
function d = EuclideanDistance(x,y)
d = sqrt(sum((x-y).^2));
```





Appendix B. All Six Gears – Five-Record Matrix Profile Index (MPI) "Snapshot" Plots for Propagation and Baseline Runs

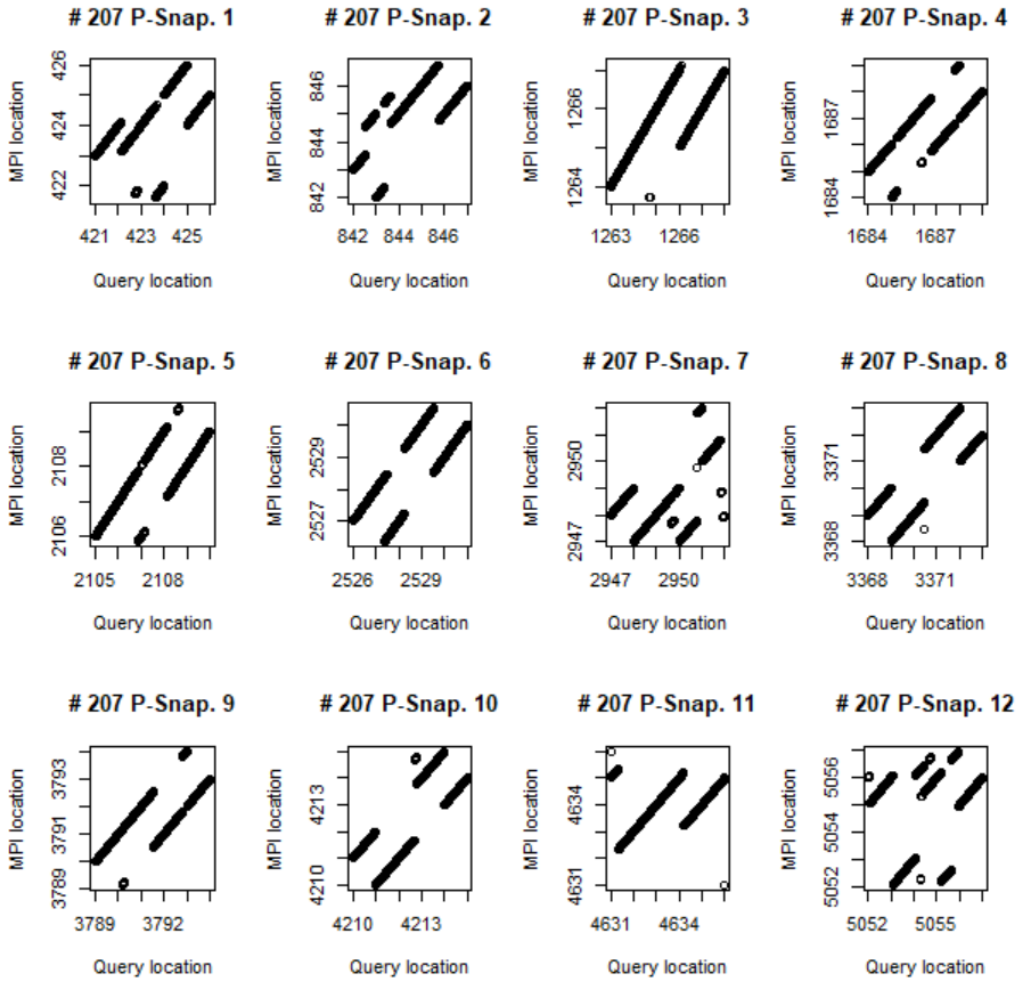


Fig. B-1 Gear 207 propagation data. Data snapshots 1–12: MPI plots.

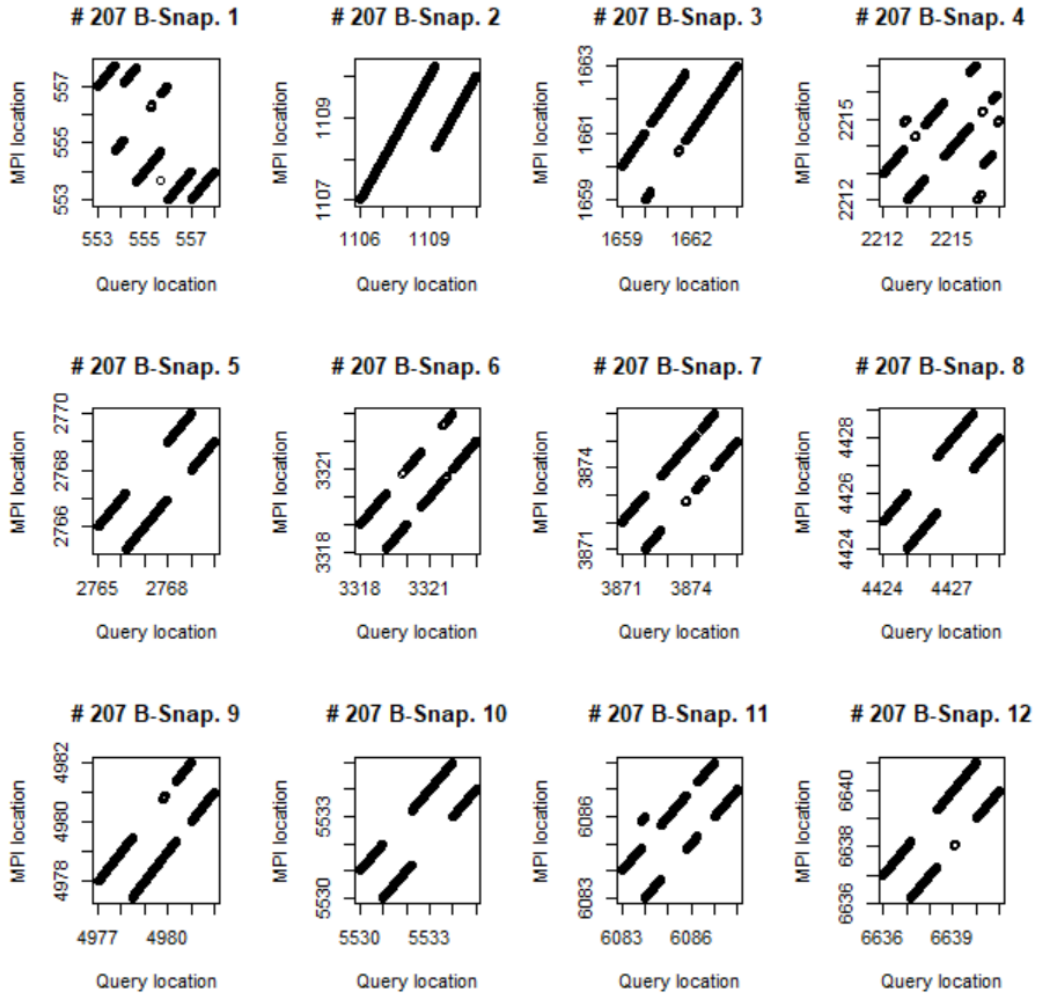


Fig. B-2 Gear 207 baseline data. Data snapshots 1–12: MPI plots.

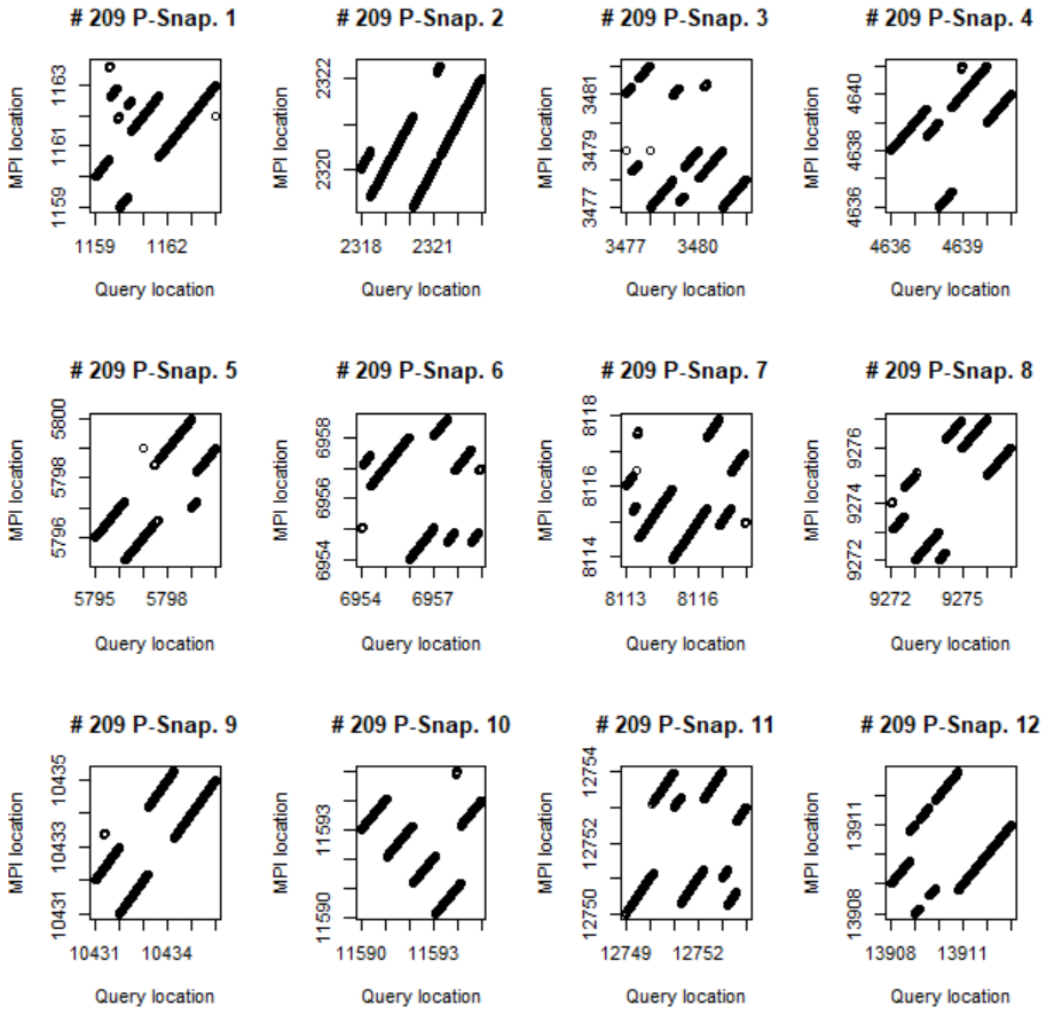


Fig. B-3 Gear 209 propagation data. Data snapshots 1–12: MPI plots.

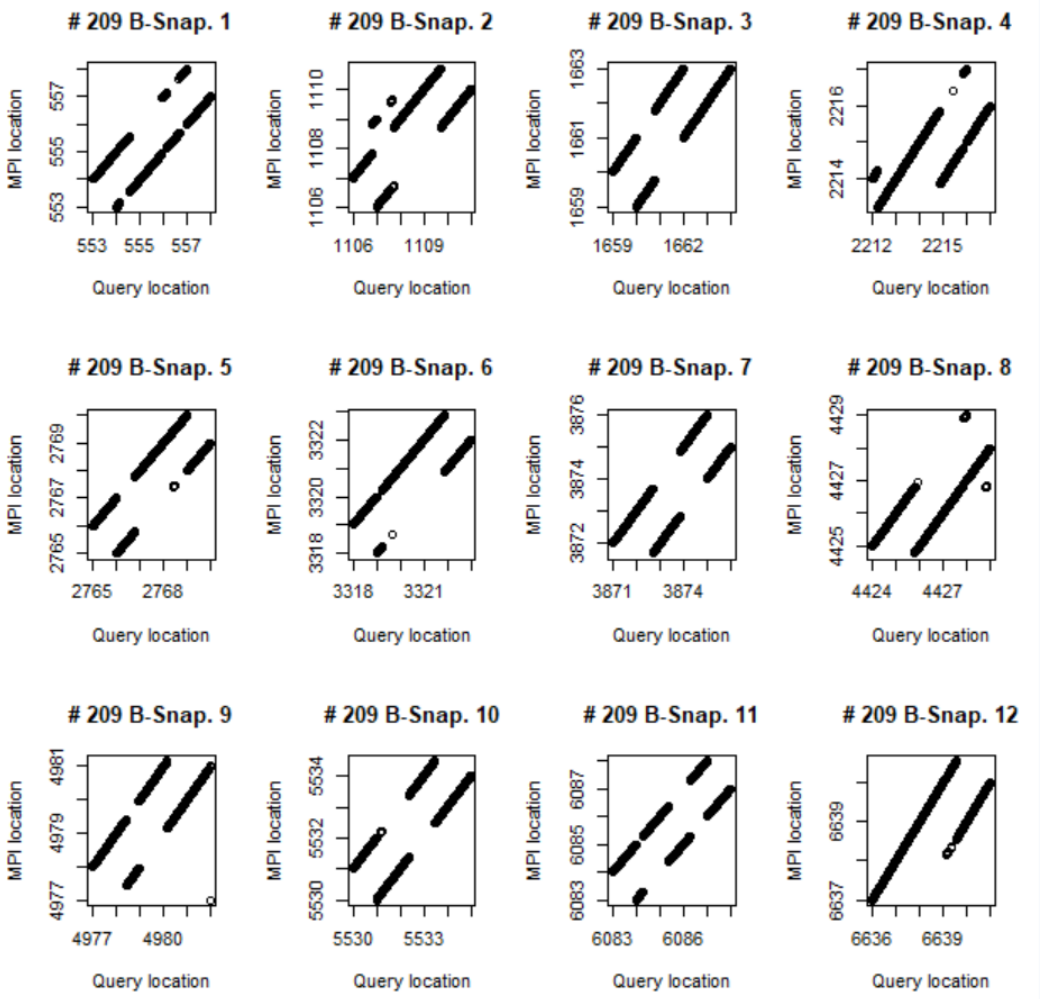


Fig. B-4 Gear 209 baseline data. Data snapshots 1–12: MPI plots.

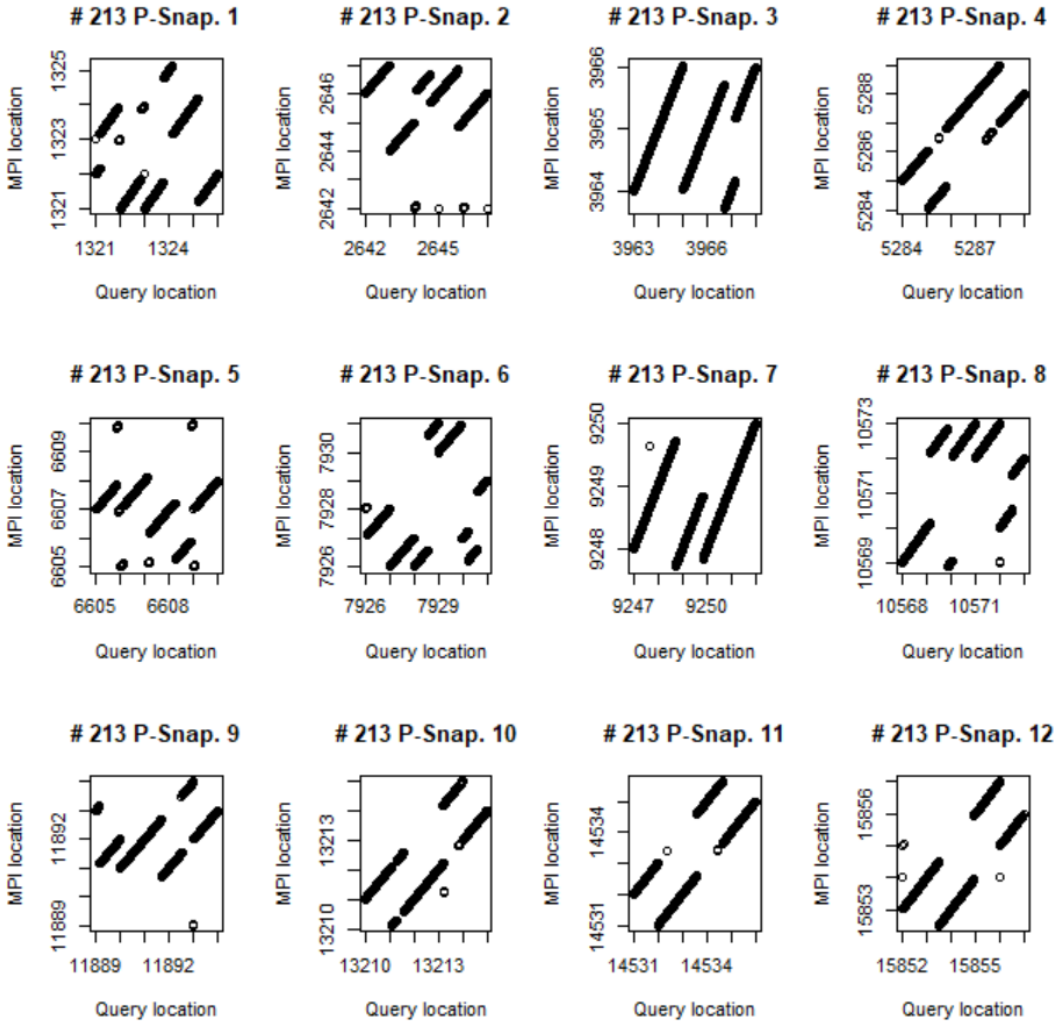


Fig. B-5 Gear 213 propagation data. Data snapshots 1–12: MPI plots.

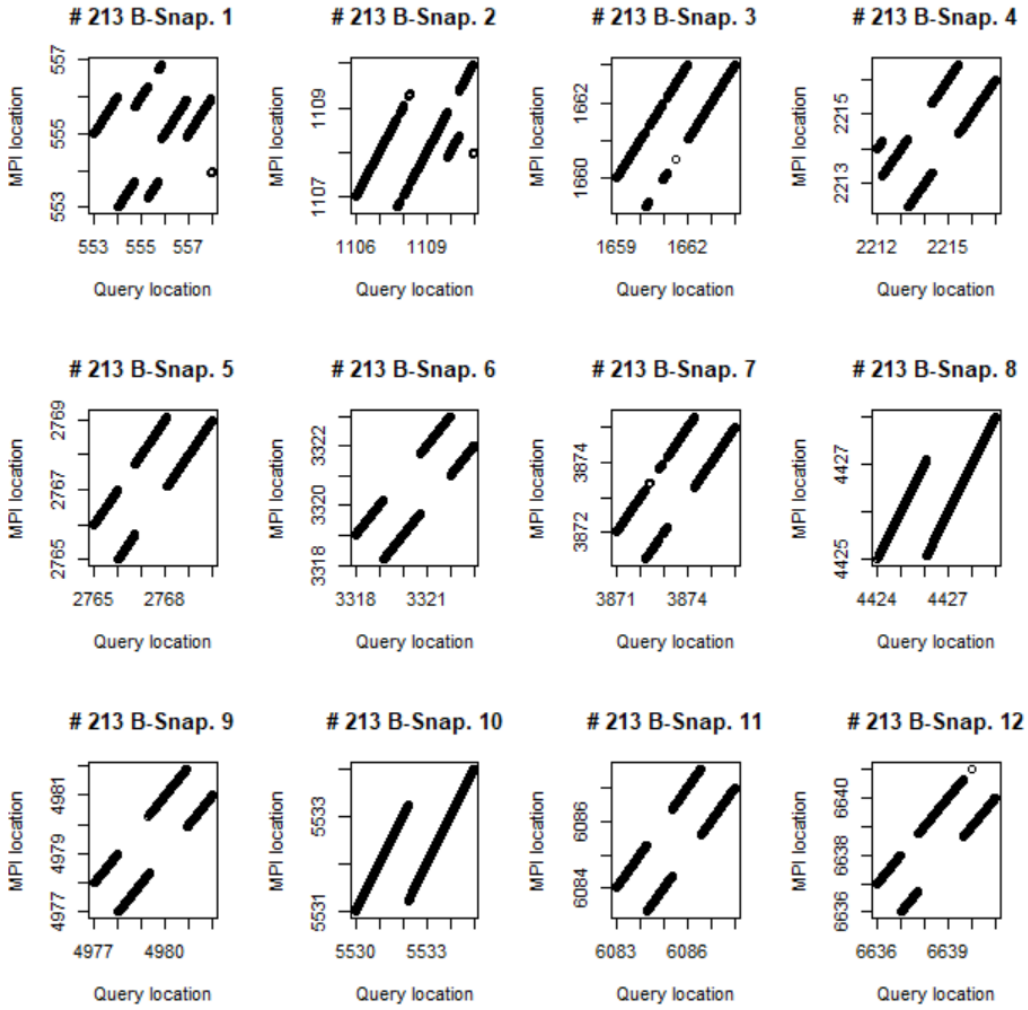


Fig. B-6 Gear 213 baseline data. Data snapshots 1–12: MPI plots.

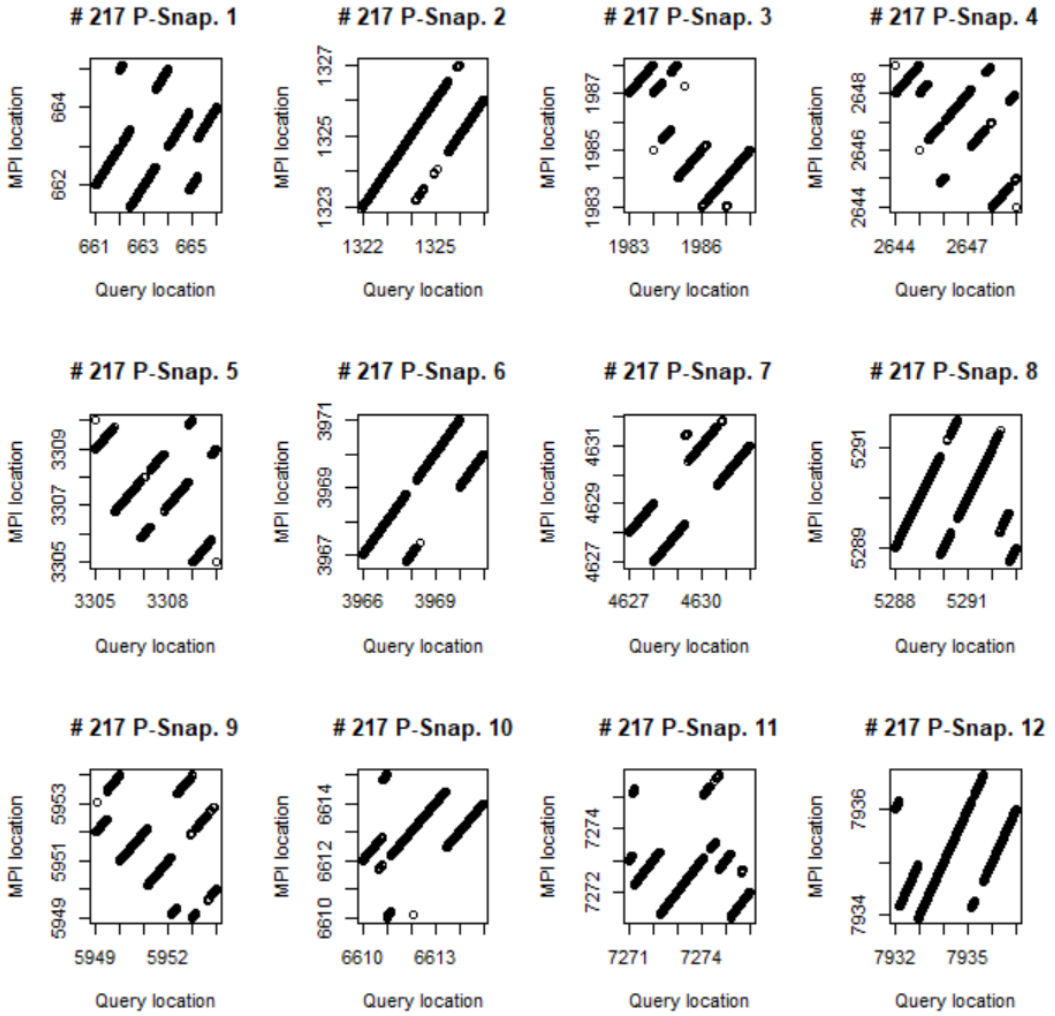


Fig. B-7 Gear 217 propagation data. Data snapshots 1–12: MPI plots.

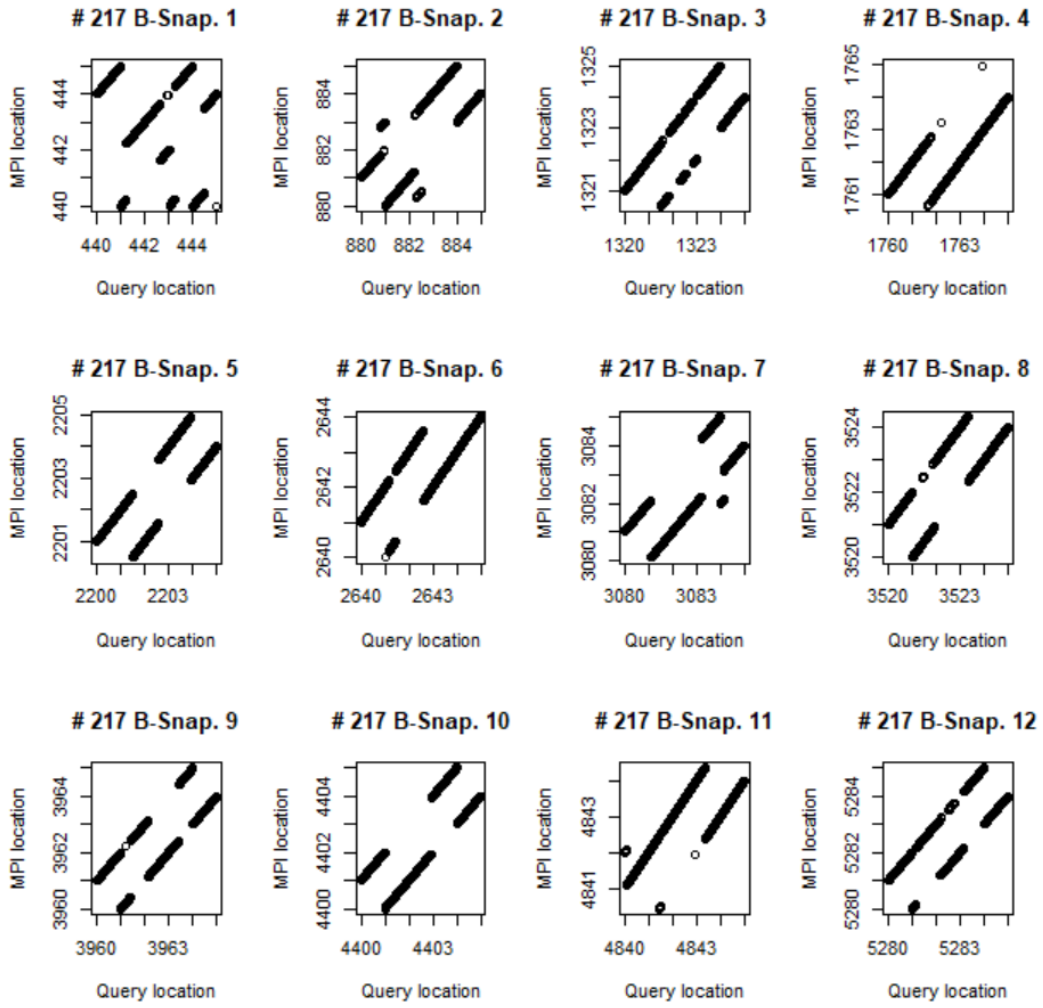


Fig. B-8 Gear 217 baseline data. Data snapshots 1–12: MPI plots.

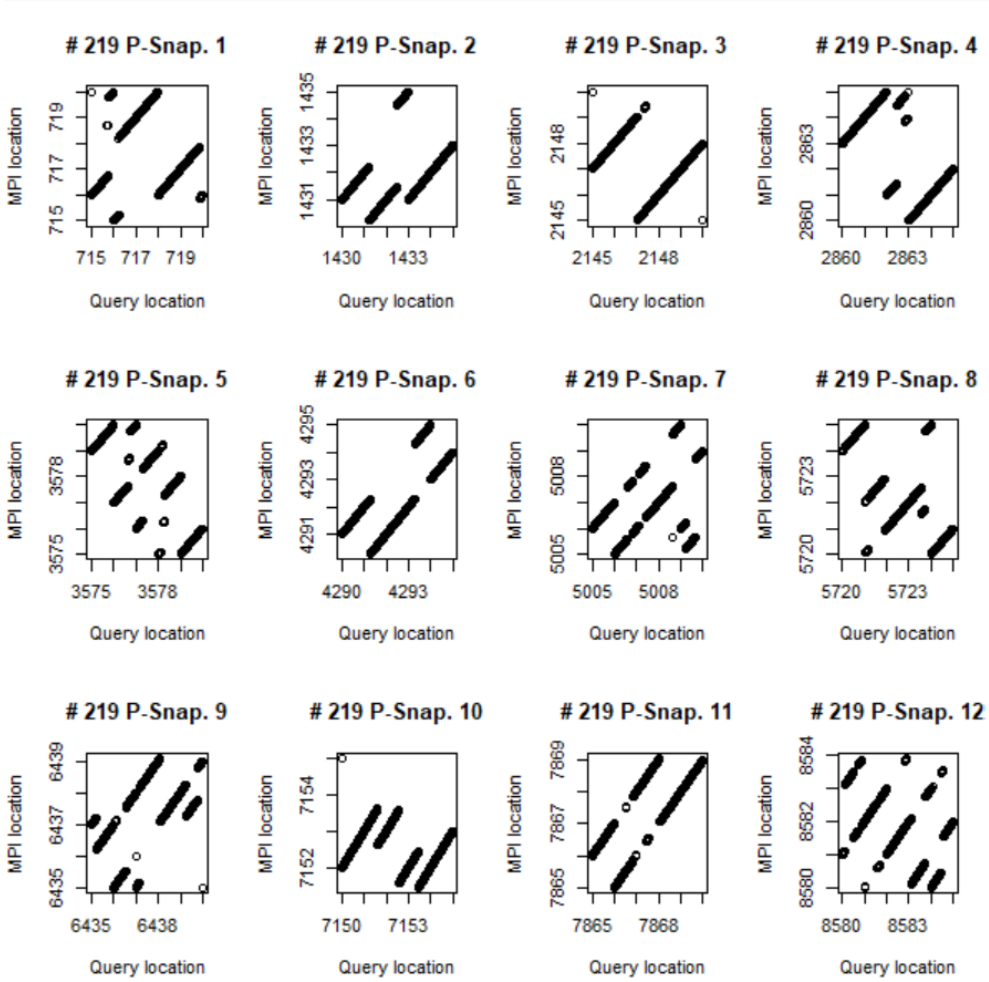


Fig. B-9 Gear 219 propagation data. Data snapshots 1–12: MPI plots.

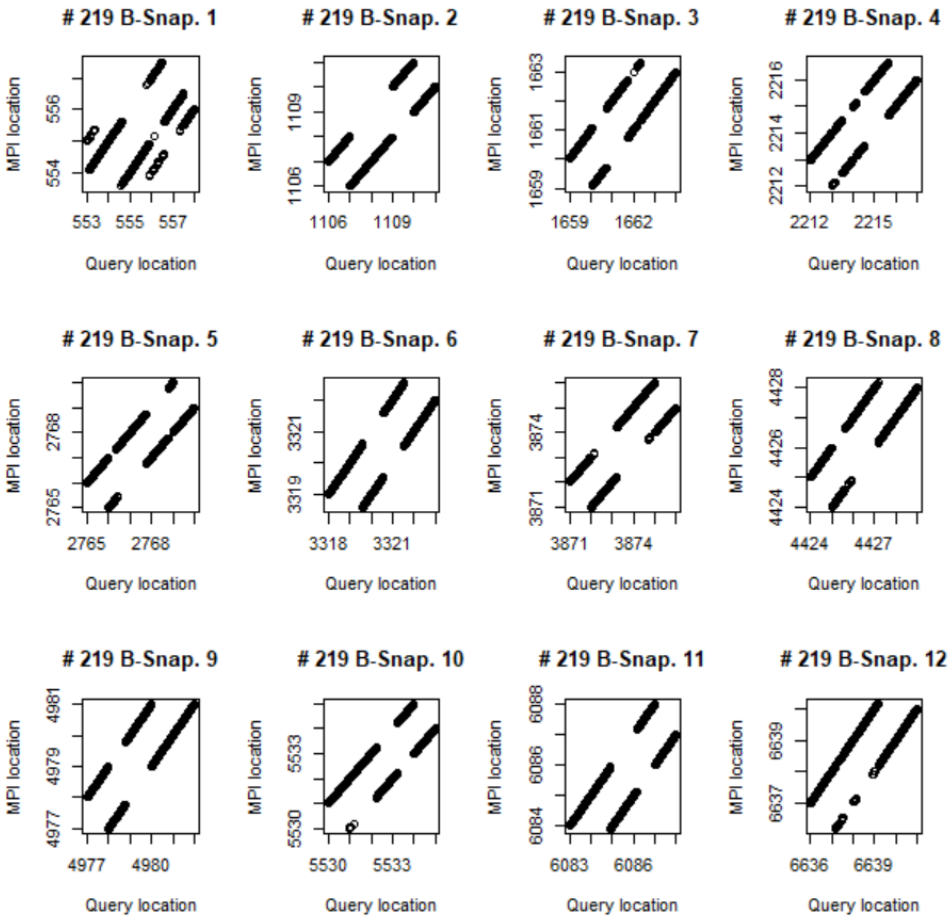


Fig. B-10 Gear 219 baseline data. Data snapshots 1–12: MPI plots.

Appendix C. Five-Record Matrix Profile Index (MPI) Variances Sorted to Show Number of Zero Values

Data row	Five-record snapshots: Baseline MPI variances (sorted lowest to highest)									
[1]	0	0	0	0	0	0	0	0	0	0
[11]	0	0	0	0	0	0	0	0	0	0
[18]	0	0	0	0	0	0	0	0	0	0
[31]	0	0	0	0	0	0	0	0	0	0
[41]	0	0	0	0	0	0	0	0	0	0
[51]	0	13107	21272	123594	129242	179891	253285	422941	583545	695909
[61]	776984	822676	926889	1622781	1728969	2580926	2654658	5377696	7906919	11253396
[71]	27290895	31694458								

Data row	Five-record snapshots: Propagation MPI variances (sorted lowest to highest)									
[1]	0	0	0	0	0	0	0	0	0	0
[11]	0	0	819	14744	83953	191103	201190	311828	369933	706376
[18]	1272261	1778157	2038122	2416055	3062255	3142089	3528196	3541838	3903940	4093199
[31]	4117395	4194509	5767711	6409698	6446498	6701491	6883487	7144372	7476312	7697499
[41]	8330235	8364016	8468410	8890434	9481987	9887485	10415919	10482254	11096837	12106207
[51]	12187270	12245831	12512918	13372638	13536423	14066251	14347842	14386715	15030260	15312484
[61]	16174336	18925571	20907849	21388032	21716033	24698997	26350762	27949593	32622174	33900526
[71]	34178676	35486289								

List of Symbols, Abbreviations, and Acronyms

AI	artificial intelligence
AUC	Area Under the Curve
CI	Condition Indicator
ED	Euclidean Distance (z-normalized)
MPI	Matrix Profile Index
MSE	mean square error
NNET	neural net
PHM	prognostic health management
SP	search span
TR	reference time series
TSA	time-synchronous averaging
Q1	query (usually a short segment of a time series)

1 (PDF) DEFENSE TECHNICAL INFORMATION CTR DTIC OCA

1 (PDF) DEVCOM ARL FCDD RLD DCI TECH LIB

9 (PDF) DEVCOM ARL FCDD RLC IS
A HURWITZ
C LENNON
K SCHAEFER-LAY
D BARAN
B MAC CALL
FCDD RLW VA
R EMERSON
A HOOD
FCDD RLW VB
M HAILE
A HALL