INTERPRETING CHANGE IN TRANSMISSION PIPELINE CORROSION FROM IN-LINE INSPECTION INSTRUMENT DATA

UNDERGRADUATE HONORS THESIS

By

Andrew R. Lutz

The Ohio State University

Department of Mechanical Engineering

23 February 2007

Advised By:

Dr. Marcelo J. Dapino, Advisor:

Table of Contents

1. Introduction and Objectives

According to the American Petroleum Institute, 65% of our nation's energy is supplied by oil and natural gas. Pumping it through pipelines is how much of this oil and natural gas is transported. Today, of the 300,000 miles of natural gas transmission pipelines in the U.S., 62% were built between 1940 and 1970 (Clark, Leis, and Eiber 2004). If this aging infrastructure were to fail, it would be crippling to society.

One of the largest problems plaguing pipelines is corrosion. According to a National Association of Corrosion Engineers federal study, "corrosion costs U.S. transmission pipelines as much as 8.6 billion dollars per year" (Thompson and Vieth, 2003). Tools known as in-line instruments (ILI's) have capabilities to identify and predict the size of corrosion patches on pipelines. Data sets from in-line instruments can contain data from hundreds of miles of pipe. If an accurate and economical method could be found to quantify changes in corrosion data over time, this could prove useful towards predicting the life of pipelines. The current research has determined such a method. The results of this method have the capabilities to predict and quantify corrosion growth. With this information, a pipeline owner/operator will be better equipped to determine necessary repairs within their system and curb part of the huge corrosion costs associated with regular maintenance and catastrophic failure (with its associated cost in terms of loss of life and litigations).

2. In-Line Instrument Background

When inspecting a pipeline there are two major physical limitations. First, pipelines are often buried, requiring one to dig into the ground if one wants to observe them directly. The second limitation lies in the fact that pipelines can be hundreds of miles long. With this type of length even an unburied pipeline would take a person weeks or months to observe directly. To solve these problems, the pipeline industry employs tools called in-line instruments (ILI's), also known as Pigs or Smart Pigs. In-line instruments are self contained tools that are sent down the center of a pipeline to take measurements about the pipe wall (Bubenik, Nestleroth, and Leis 2000). They are designed to measure many specific aspects about a pipe wall. Features that are detectable by in-line instruments include corrosion, mill defects, cracks, dents, welds, bends, valves, and repairs among other things.

There are a number of different technologies being used by ILI's today. Three of the most common ILI technologies are magnetic flux leakage detection, ultrasonic detection, and caliper techniques. ILI caliper tools use the simplest measurement principle. An ILI caliper tool is shown in figure 2-1.

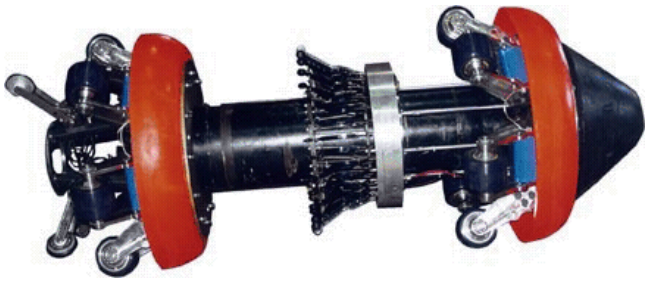


Figure 2-1: In-Line Instrument Caliper Tool

Note: BJ Services GEOPIG Caliper Tool

It is noted from the picture that ILI caliper tools have arms known as calipers extending to the inner surface of the pipe wall. These extensions occur around the entire inner circumference of the pipe. The purpose of these calipers is to measure the inner radius of the pipe at regular circumferential intervals. As the tool is sent down the length of the

pipe, radial measurements are taken by each arm and recorded. If there is a dent in the pipe, one or more of the calipers provide a measure of the change in the pipe radius at the location of the dent. This is a commonly used method of determining if there are any dents, bends, or other geometry changes in a pipeline.

Another measurement principle found in ILI's is magnetic flux leakage (MFL) detection. Magnetic flux leakage tools exploit the properties of magnetic flux to measure different aspects of a pipe wall. Corrosion metal loss is one type of feature that can be detected by a magnetic flux leakage ILI (pictured below).

Figure 2-2: In-Line Instrument Magnetic Flux Leakage Tool

Note: Baker Hughes CPIG HR20

The magnetic flux leakage tool has two primary components. First, the tool contacts the pipe wall with strong magnets causing magnetic flux through the pipe wall. The second component is an instrument that measures the magnetic field near the pipe wall. A diagram showing magnets creating magnetic flux through a circumferential pipe wall section is shown below.

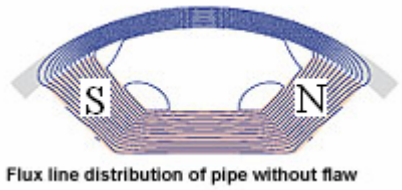


Figure 2-3: ILI Applies Strong Magnets to the Pipe Wall Causing Magnetic Flux within the Wall

Note: Picture from Tuboscope Website

When the tool passes by an area of metal loss within the pipe wall, the magnetic flux leaks out of the wall material. The flux leakage causes a change in the magnetic field close to the pipe wall at the location of the metal loss. This change in the magnetic field is then measured and recorded. A picture depicting the change in magnetic field due to metal loss is shown in Figure 2-4.

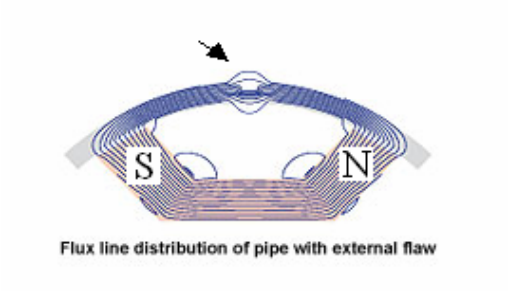


Figure 2-4: Metal Loss within the Pipe Wall causes the Magnetic Flux to Leak Changing the Proximal Magnetic Field

Note: Picture from Tuboscope Website

The signal data from the magnetic flux leakage is recorded by the instrument and then extracted at a later time for analysis. Features such as corrosion metal loss, gouges, and mill defects each have a distinctive signal signature. When the signal is analyzed, prediction of the size and location of each detected feature is made. Figure 2-5 shows raw signal data with a distinct flux leakage occurring in the highlighted region. This is how features on the pipe wall are detected.

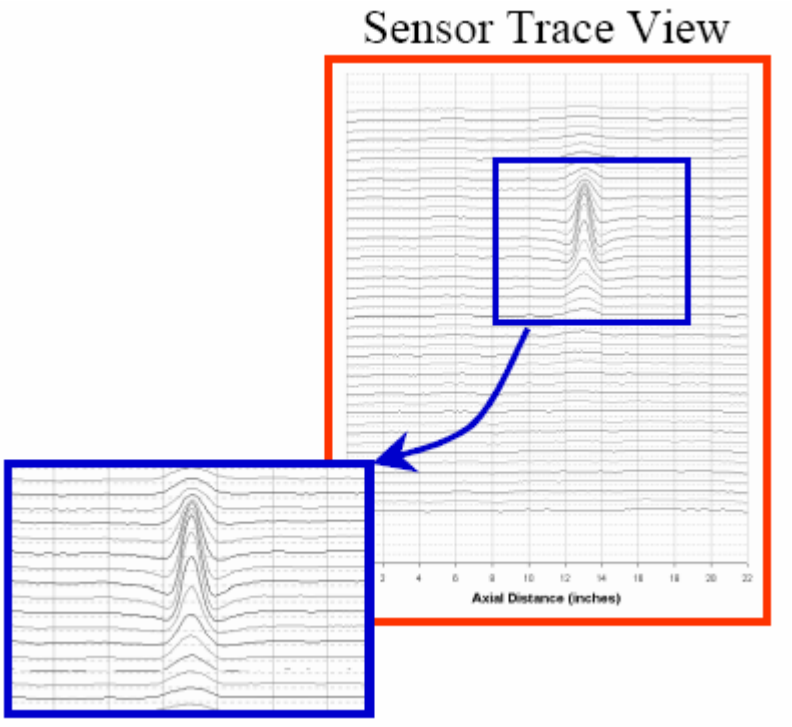


Figure 2-5: Magnetic Flux Leakage Signal Data
with a Highlighted Increase in Flux Leakage
Note: Picture from Bubenik, Nestleroth, and Leis 2000

The third technology that is commonly utilized by in-line inspection tools is ultrasonics. An Ultrasonic ILI is shown here.

Figure 2-6: Ultrasonic ILI with a Close-Up of the Instrumentation.
Note: Tuboscope's UT Tool

In general, Ultrasonic ILI's gather information by emitting an ultrasonic wave in the direction of the pipe wall. This type of ILI also carries an ultrasonic probe which is placed near the pipe wall to measure the wave's echo when it is returned. This measurement principle is being applied to crack detection in Figure 2-7.

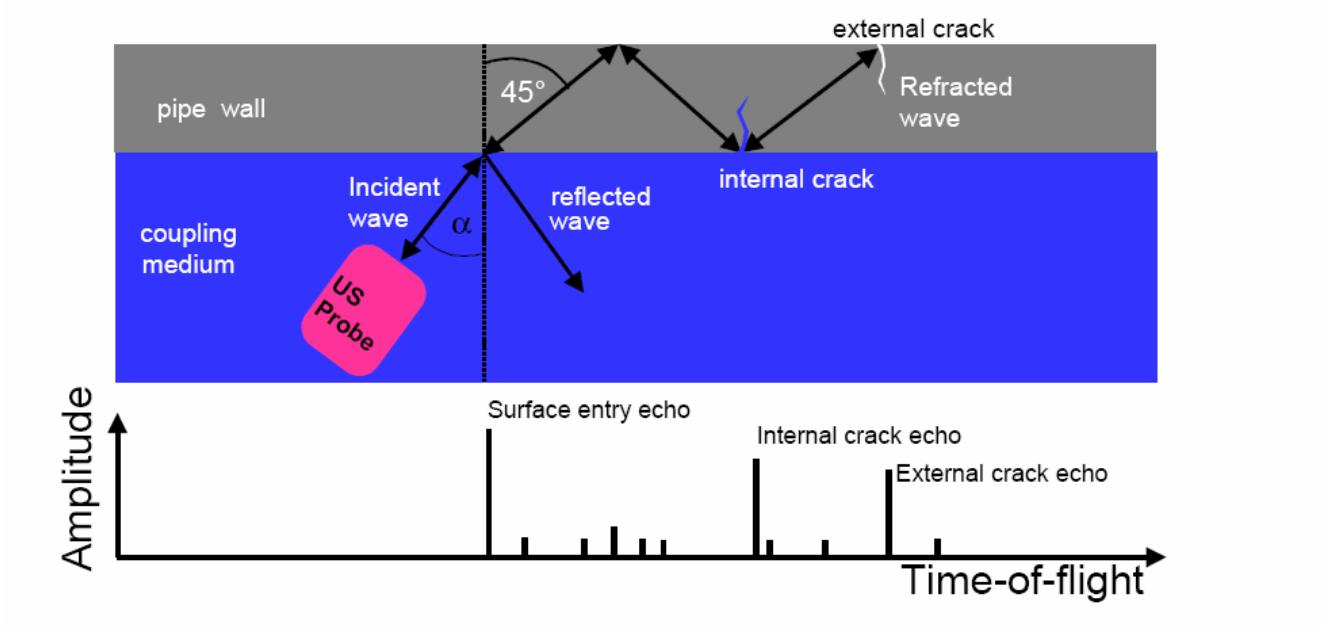


Figure 2-7: Ulrasonic Measurement Principle

Note: Picture from Reber and Beller 2003.

Figure 2-7 shows a part of an axial cross section of a pipe containing an ultrasonic ILI. The ultrasonic probe is depicted near the pipe wall. The ultrasonic wave is shown entering the pipe wall at an angle in such a way that it travels axially down the pipe. As the wave passes through cracks in the pipewall, signature echoes are returned to the probe allowing the location and size of the cracks to be recorded. Corrosion metal loss can be measured using the same general measurement principle. However, to measure corrosion metal loss the instrument configuration is slightly different. Ultrasonic ILI's report corrosion metal loss with predictions for its size and location on the pipe.

Each of the three ILI's mentioned has its own strengths and weaknesses. Both the magnetic flux leakage and ultrasonic can be used to measure corrosion metal loss in pipelines. The corrosion metal loss data is typically processed into a format called a pipeline listing. Pipeline listings from ultrasonic and magnetic flux leakage tools appear

similarly. Corrosion metal losses are reported with a predicted size and location by both ILI types. Since different types of in-line instruments produce corrosion data in a pipeline listing the approach of the current research has been designed to analyze this data format. The purpose of this design is so this approach can be applied to data from different types of in-line instruments.

3. Approach

3.1 Corrosion Metal Loss Matching in a 2-D Wrapping Plane

When in-line instruments (ILI's) report a corrosion metal loss feature, there are predictions made about the feature's size and location. The location prediction includes axial and circumferential components. The axial location is typically reported as a distance measured from a known reference point. The circumferential location is typically reported as an O'clock value or a location in degrees from a reference orientation on the pipe's circumference. The sizing predictions are generally made in terms of axial length, circumferential width, and radial depth of the corrosion metal loss. All of this data is usually reported in an item known as a pipeline listing. An abbreviated pipeline listing showing data for three corrosion metal loss features is shown in table 3-1.

Table 3-1: Abbreviated Pipeline Listing

Pipeline Listing					
PipeLine Feature	Odometer (Feet)	O'Clock Orientation	Predicted Length (in)	Predicted Width (O'Clock)	Predicted Depth (% of Wall Thickness)
Metal Loss	791727.1	6:14	6.1	1:55	12%
Metal Loss	791727.8	7:40	7.6	2:24	19%
Metal Loss	791729.6	10:48	4.3	2:02	11%

One goal of predicting the size and location parameters is to help in determining what parts of the pipe are affected most by corrosion.

The goal of the current research is to compare two ILI data sets acquired from one pipeline at different times. This comparison will be used to quantify the changes and differences between the corrosion metal losses measured in each ILI data set. An algorithm has been developed to find corrosion metal loss features, which were measured to have occurred at the same location. The method in this algorithm is to model each corrosion metal loss feature as a rectangular area with a length and width equal to the respective values predicted by the ILI. The center of this rectangle, which may also be called a box, is located at the point identified by the ILI as the corrosion metal loss

location. These corrosion rectangles or boxes can be plotted in a two-dimensional (2-D) plane. The x-axis in this 2-D plane represents the axial odometer and the y-axis is the O'clock orientation. When the 2-D plane is created in this way it is as if the surface of the pipe were cut axially along the 12:00 orientation line, unwrapped, and placed flat on a plot. For this reason, this 2-D plane will be titled a wrapping 2-D plane. Figure 3-1 shows the rectangles from the abbreviated pipeline listing of Table 3-1 being plotted in a wrapping 2-D plane.

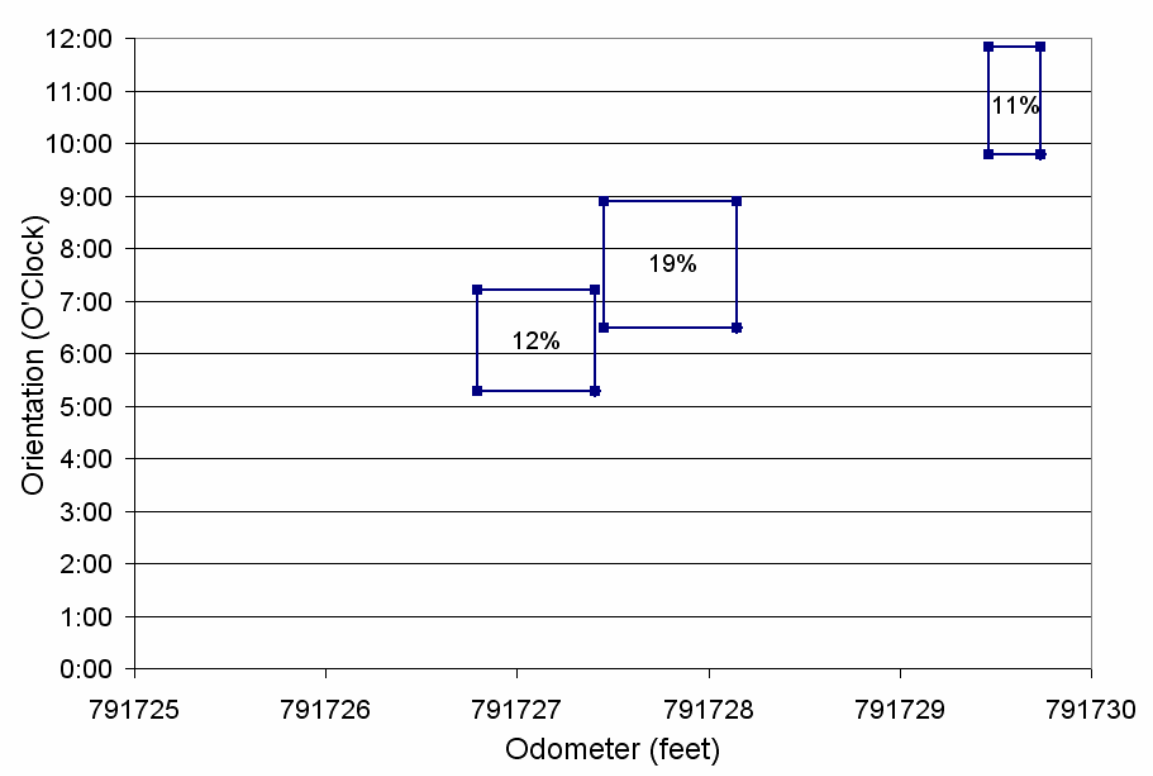


Figure 3-1: Corrosion Rectangles Plotted in 2-D Wrapping Plane

According to this algorithm, each corrosion box is defined by its corner points. This is done because it was found that the corner points of two rectangles plotted in a 2-D wrapping plane could be used to determine if the boxes are overlapping. Overlapping boxes are defined in this algorithm as occupying any amount of common space in the 2-D wrapping plane. An example of two overlapping boxes is in Figure 3-2.

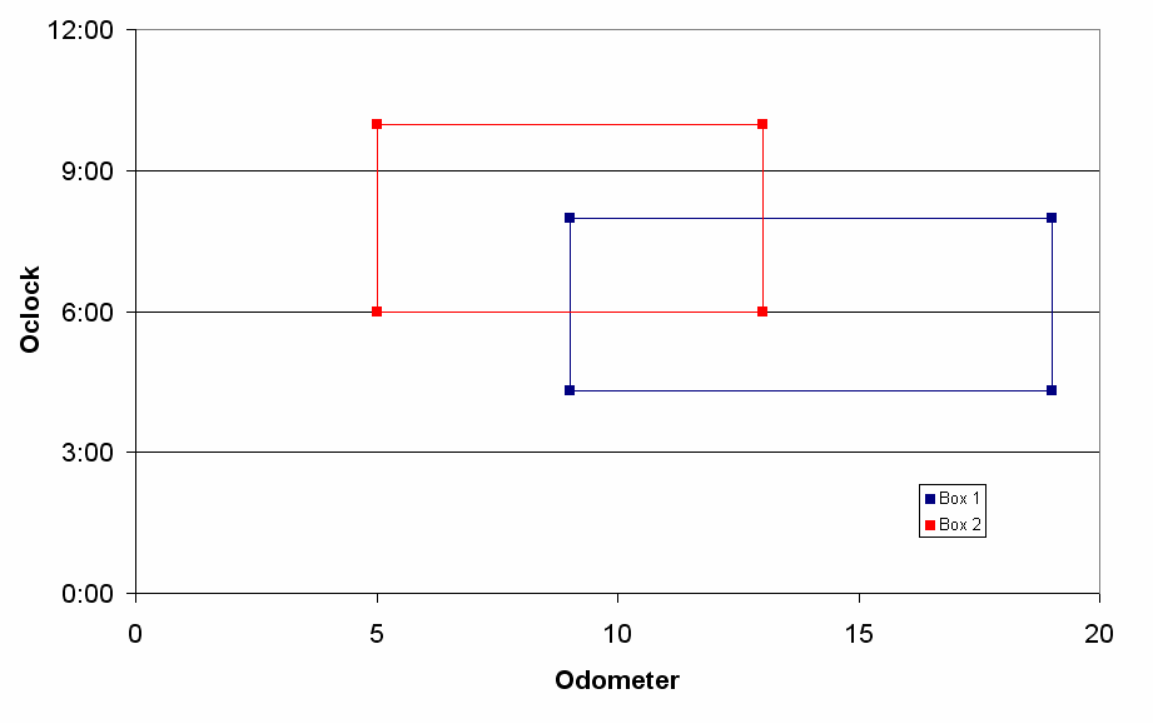


Figure 3-2: Overlapping Corrosion Boxes Plotted in a 2-D Wrapping Plane

Using the method described herein, it is possible to determine if any two boxes are overlapping. This means that it is possible to determine if two corrosion metal loss rectangles are overlapping. If corrosion metal loss rectangles from two ILI data sets taken from the same pipeline at different times are plotted on the same 2-D wrapping plane, it can be determined when a rectangle from one data set is overlapping with a rectangle from the other data set. One major assumption the current research is based on is the fact that it is reasonable to assume that when two corrosion rectangles from ILI data sets taken from the same pipeline are overlapping, they are representative of the same physical corrosion on the pipeline.

The example of rectangle overlap shown in Figure 3-2 is the simplest example encountered in a 2-D wrapping plane. The situation becomes more complicated if one of the rectangles is crossing the 12:00 orientation line. If this happens, the rectangle is split

on the 2-D wrapping plane. Part of the rectangle exists at the top of the plane while the other part sits at the bottom of the plane. This is because the 12:00 orientation line and the 0:00 orientation line in a wrapping 2-D plane represent the same line in space. The split of a rectangle across the 12:00 orientation line changes the relationship between the corner points of two overlapping rectangles. Recall that this algorithm has defined each rectangle by the position of its corner points. Because this has an effect on corner point position, it is necessary for the algorithm to consider whether or not a rectangle crosses the 12:00 orientation line as a defining parameter of each rectangle. An example of two boxes overlapping when one is crossing the 12:00 orientation line is shown below in Figure 3-3.

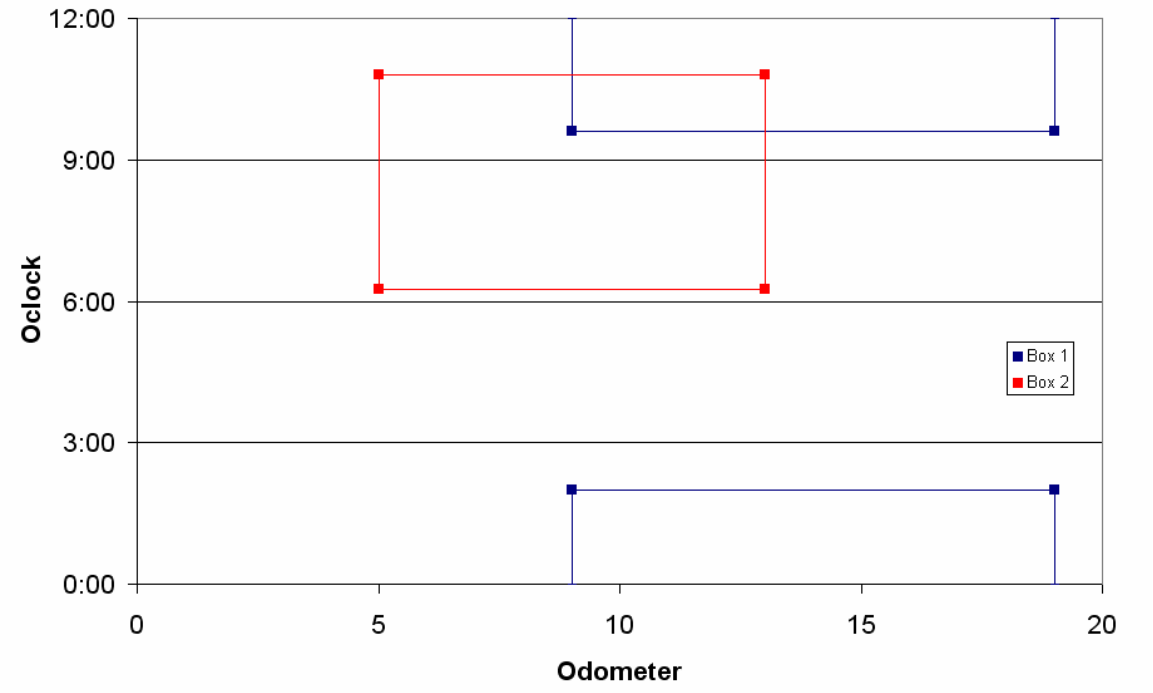


Figure 3-3: Overlapping Corrosion Boxes with One Box Crossing the 12:00 Orientation Line

The case shown in Figure 3-3 and the case shown previously in Figure 3-2 make two distinctly different cases of overlap. There are three more cases similar to the one shown above, in which one of the two rectangles are crossing the 12:00 orientation line. These cases are shown in Figures 3-4, 3-5, and 3-6.

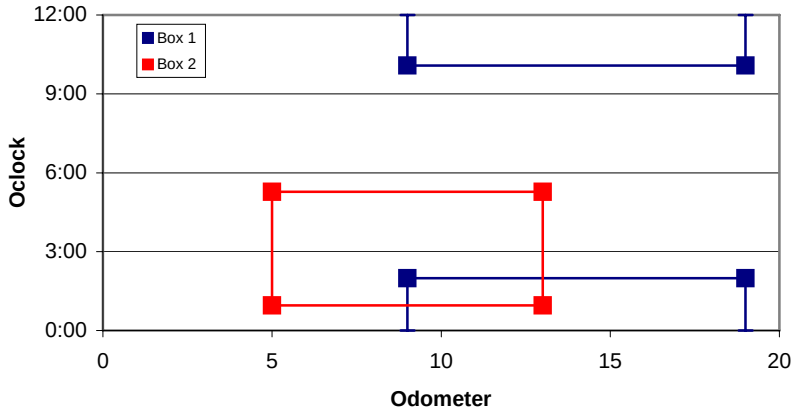


Figure 3-4: Overlapping Corrosion Boxes with a 12:00 Orientation Cross

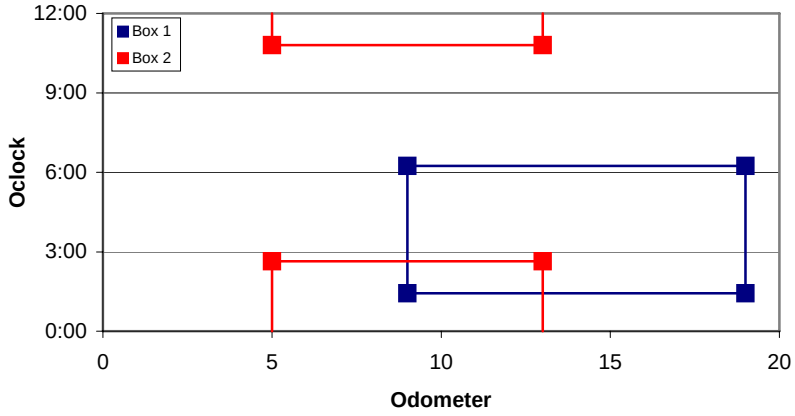


Figure 3-5: Overlapping Corrosion Boxes with a 12:00 Orientation Cross

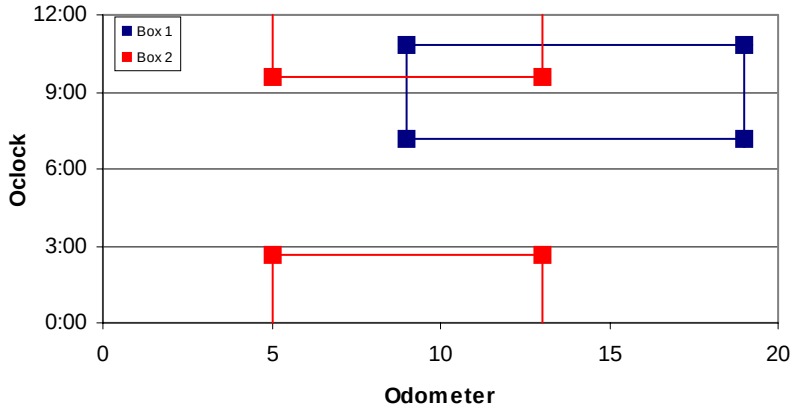


Figure 3-6: Overlapping Corrosion Boxes with a 12:00 Orientation Cross

The differences between these three cases and that of Figure 3-3 are subtle. However, the differences are significant enough to change the locations of the two rectangle's corner points relative to one another. This is why they are presented as different cases of overlap.

A sixth case of rectangle overlap occurs when both rectangles are crossing the 12:00 orientation line. Again, this case presents a situation where the relative corner locations of the two rectangles are different from any of the previous cases. This case is shown below in Figure 3-7.

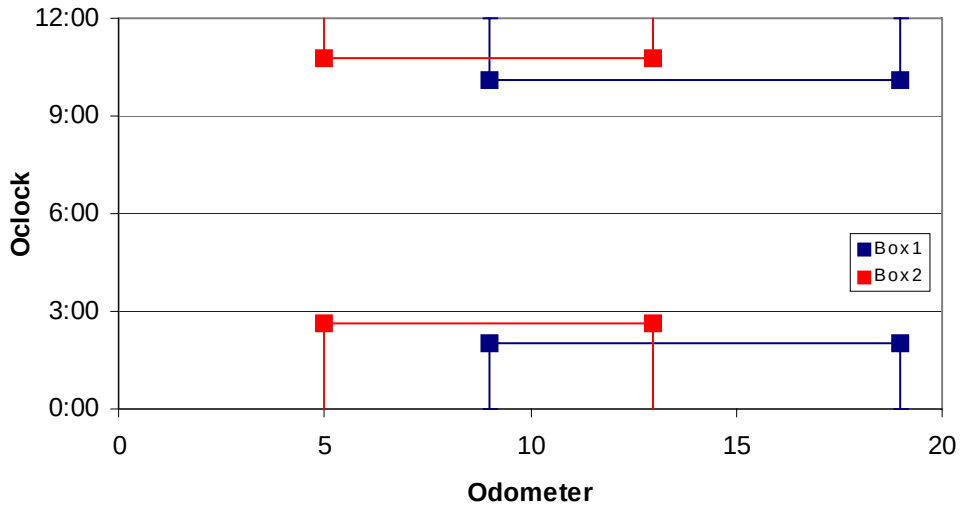


Figure 3-7: Overlapping Boxes which are both Crossing the 12:00 Orientation

There is one additional parameter, which the algorithm uses to define a rectangle in a 2-D wrapping plane. This parameter occurs when the rectangle has a width of the entire circumference of the wrapping plane. This width is equivalent to twelve hours on the O'clock y-axis. This case has been titled a 360 degree rectangle due to its 360 degree nature. There is a series of rectangle overlap cases involving 360 degree rectangles. These cases are shown below in Figures 3-8 and 3-9.

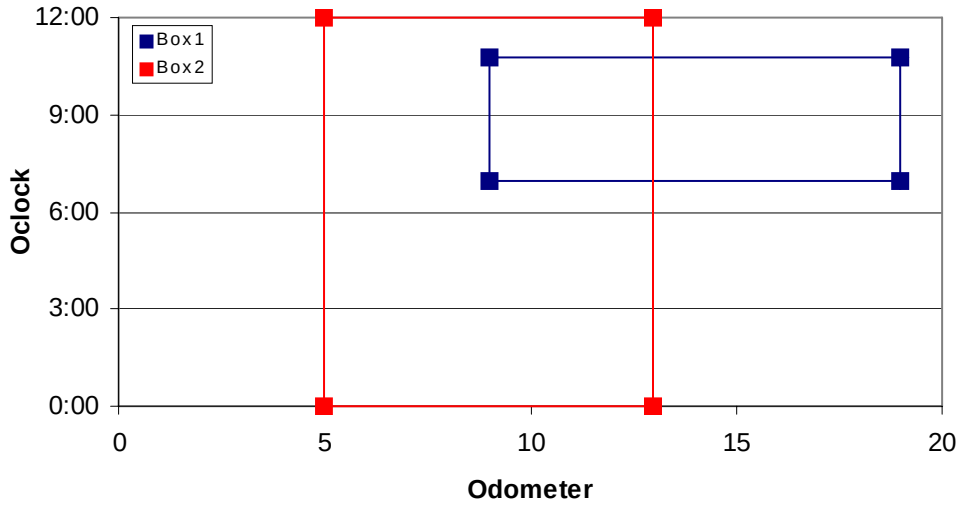


Figure 3-8: Overlapping Boxes When One has a 360 Degree Width

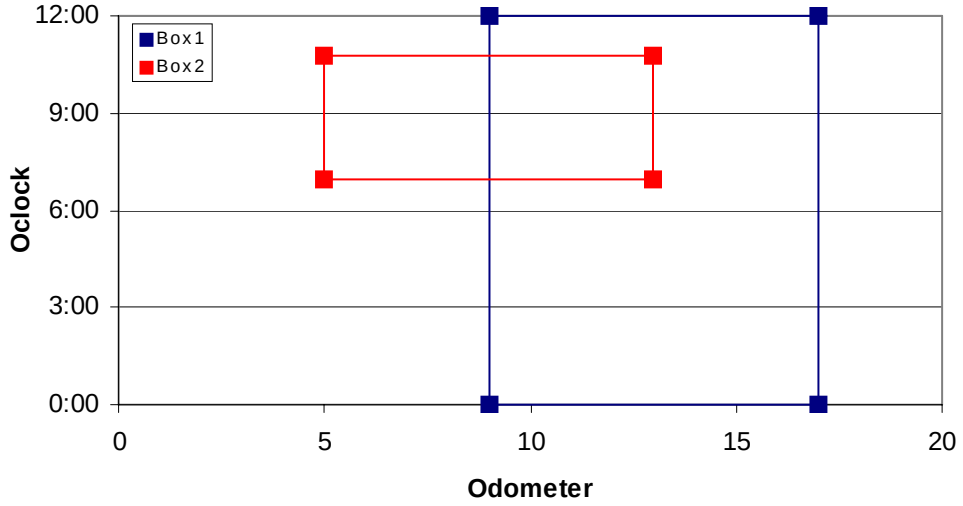


Figure 3-9: Overlapping Boxes When One has a 360 Degree Width

In addition to what has been shown so far there are four more cases of rectangle overlap. These cases involve rectangles that are crossing the 12:00 orientation line. These cases are different from those previously presented because in each of these cases, the rectangles are overlapping in such a manner that they combine to cover the entire circumference of the pipe. These additional cases are shown her in figures 3-10, 3-11, 3-12, and 3-13.

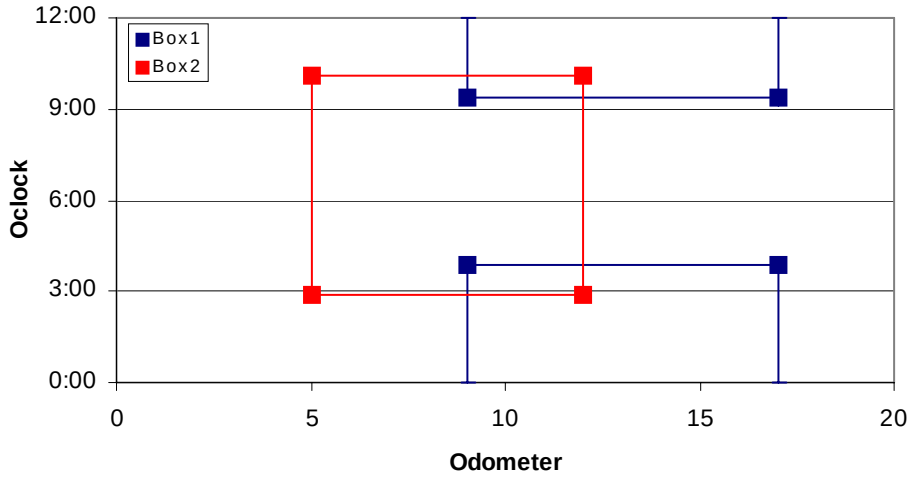


Figure 3-10: Overlapping Boxes that Combine to Cover 360 Degrees

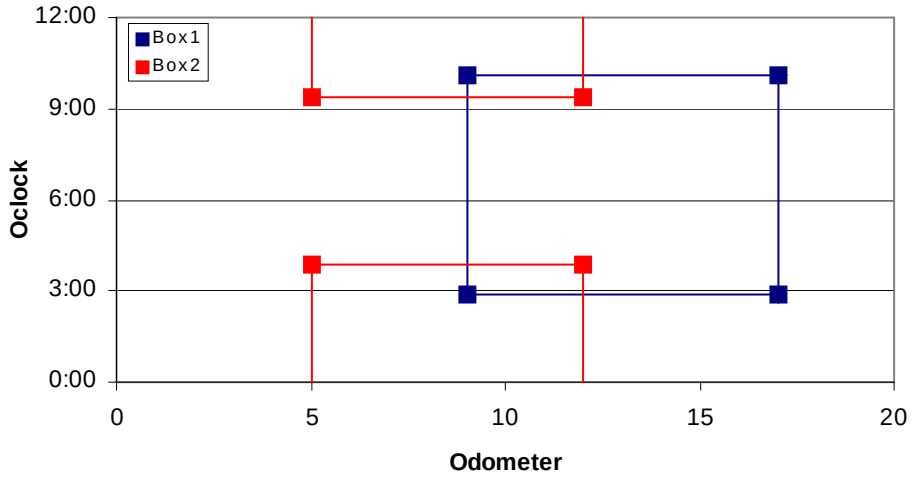


Figure 3-11: Overlapping Boxes that Combine to Cover 360 Degrees

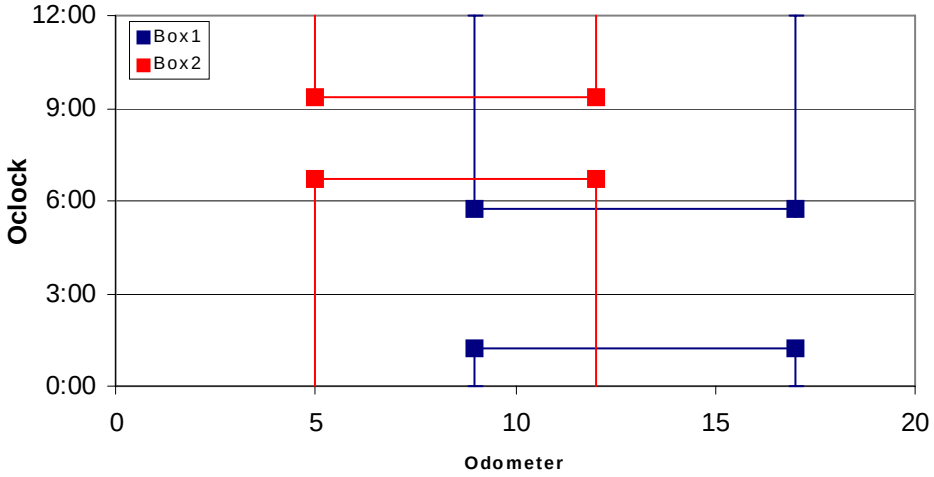


Figure 3-12: Overlapping Boxes that Combine to Cover 360 Degrees

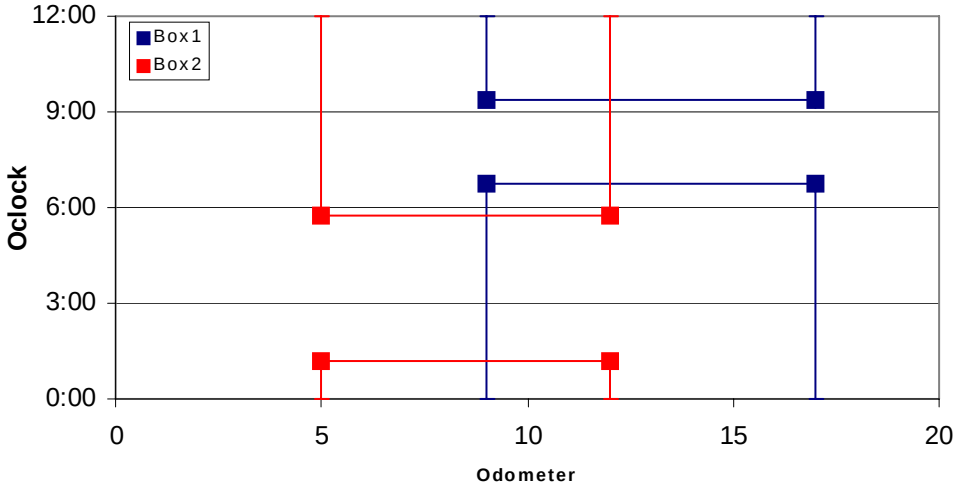


Figure 3-13: Overlapping Boxes that Combine to Cover 360 Degrees

Including the cases in these four figures, there are a total of twelve cases. These cases are assumed to encompass all possible instances of two rectangles overlapping in a 2-D plane. Each of these cases also contains rectangles with unique sets of defining parameters. This supports the idea that the same parameters that have been used to define

the rectangles in a wrapping 2-D geometry can be used to define instances of two overlapping rectangles. The parameters of relative corner point location, the crossing status of the 12:00 line, and the 360 degree box status of the rectangles for all unique overlapping cases are listed in the appendix.

When an ILI is run on a pipeline segment, it collects huge amounts of data. The number of reported corrosion features is typically in the thousands. The enormous size of these data sets presents an issue when processing these data. If one were required to analyze two ILI data sets by hand using the algorithm described previously, it would take days to complete. This analysis can be completed more quickly and more reliably when it is performed by a computer program. A computer script has been written in Matlab, which applies the algorithm to ILI data in the form of a pipeline listing. This program has the capability to match two sets of ILI data, with 5,000 corrosion metal loss calls each, to one another in under 5 minutes. The program is a companion to this research. A “read me” instruction file for this program has been included in the appendix.

3.2 Addressing In-Line Instrument Error

When comparing subsequent runs of in-line instrument (ILI) data, one of the toughest challenges was addressing the error associated with the ILI corrosion metal loss data. It is an industry standard for ILI vendors to quantify the errors associated with their instruments and make them available to their clients. A sample error table from an actual ILI vendor is shown in Table 3-2.

Table 3-2: Sample ILI Error and Specifications Table

Note: Specifications are from the GE TranScan Magnetic Flux Leakage Tool

TranScan™ Specifications	
Nominal Tool Diameter	**12 to 36 in (now)** **6 to 56 in (future)**
Seam Weld Inspection	
Features Reported	Manufacturing defects, lack of fusion, cracks in the long seam weld and cracks within two inches of weld
Detection Features - over 50mm long	Minimum depth 0.25t
Detection Features - 25-50mm long	Minimum depth 0.5t
Sizing - Detected Features	Depth accuracy ± 0.2t Length accuracy ± 1 in (25 mm)
Crack Width	Minimum 0.004 in (0.1 mm)
Pipe Body Inspection for Metal Loss/Mechanical Damage	
Features Reported	Axial metal loss, third-party damage, dents, gouges and dents with gouges or cracking
Detection - Features over 3t long	Minimum depth 0.2t
Detection - Features less than 3t long	Minimum depth 0.4t
Sizing - Detected Features **Length >3t** **Length <3t**	Depth accuracy: ± 0.15t Length accuracy: ± 0.8 in (20 mm) ± 0.4 in (10 mm)
Metal Loss Width - Minimum	Still being determined, less than 7 mm
Location Accuracy **- Axial** **- Circumferential**	 ± 8 in (0.2 m) from reference weld ± 5°
Operating Specifications	
Product	Liquid and Gas
Active Range (varies with tool size)	150 km (12 in) to 200 km (30 in)
Tool Speed	0.4 to 9 mph (0.2 to 4 m/s)
Operating Temperature	32 to 100 °F (0 to 40 °C)
Maximum Pressure	220 bar
Smallest Bend Radius	R=3xD
Wall Thickness for Full Specification	Up to 0.5 in (13 mm) for 12 in Up to 0.6 in (15 mm) for 30 in

This table indicates two major error types, which were calculated by the vendor using a confidence level of at least 80%. The first error is related to the instrument's capability to predict the size of each corrosion metal loss feature. The sizing is the instrument's prediction for length, width, and height. For the instruments indicated in Table 3-2, the sizing error is not very large. It is generally close to 1 inch in both the axial and circumferential directions. The second error is in regard to the location of the corrosion feature. Recall, the location is predicted by the instrument in the form of an odometer reading and an O'clock position. According to Table 3-2, the O'clock position is accurate within 5 degrees or 10 minutes. The odometer error was shown to be 8 inches from the nearest girth weld. Although 8 inches seems like a reasonable number for error, it is indicated that this error magnitude can occur as often as once per girth weld. Girth welds are the common connection joints that are used to hold pipe segments together. They typically occur every 40 – 80 feet on a pipeline. Since it is not uncommon for a single ILI run to exceed 1 million feet, the odometer error according to these figures would be larger than 3 miles. How can an error of this magnitude possibly be overcome? The answer is by banding. Banding is a technique commonly used in the pipeline industry. The banding process in the case of two subsequent ILI data sets consists of identifying the features within these sets that are known to occur at the same physical location. These features, called banding points, are most commonly girth welds, but could also include valves among other pipeline components. The banding process sets this banding point's odometer values equal in both data sets and distributes the odometer error throughout the length of pipe between it and the next banding point. In this manner the odometer error is effectively reset to zero at each banding point. If every girth weld is used as a banding point, then the axial location error will not exceed the 8 inches cited by the vendor. By reducing the error from 3 miles to 8 inches, banding is an effective technique to mitigate in-line instrument axial location error.

An interesting situation arises when feature sizing statistics are compared with the total error in both the circumferential and axial directions. The formula for total error is the sum of the location and sizing errors, shown below.

Formula 1: Total Error = Sizing Error + Location Error

Using this formula, the total axial error is 9 inches and the total circumferential error is 20 minutes or 10 degrees. A statistical analysis has been conducted on a sample data set of actual ILI corrosion metal loss data collected in the field. This analysis reveals how significant the ILI error is. The sample consisted of 5,347 corrosion metal loss features, which were detected on a 44 mile long segment of pipe. Histogram distributions for feature length and width from this data set are shown in Figures 3-14 and 3-15.

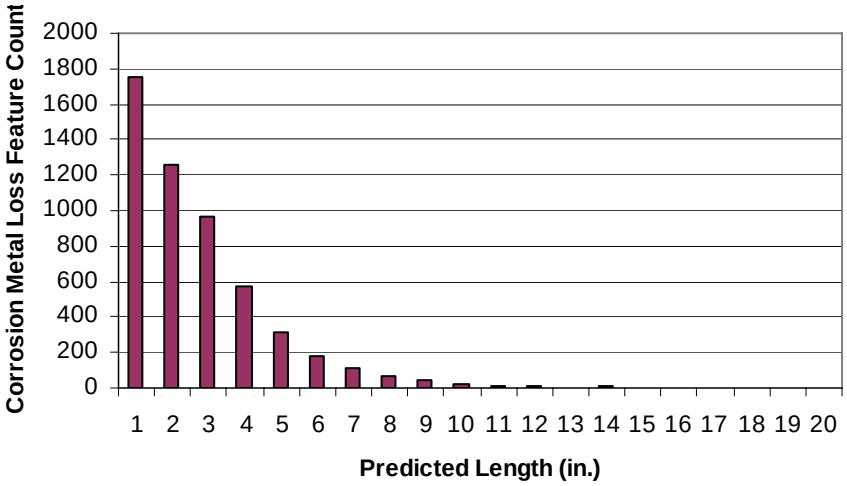


Figure 3-14: Histogram Counting Corrosion Features with Predicted Length

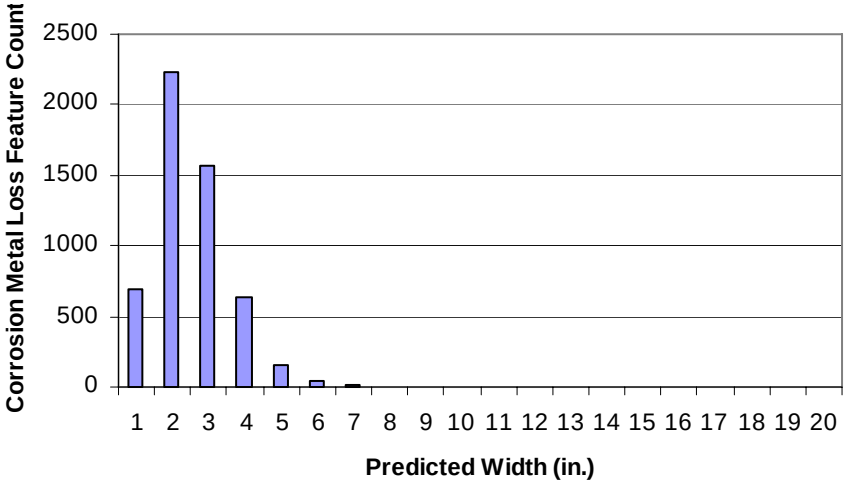


Figure 3-15: Histogram Counting Corrosion Features with Predicted Width

The average feature size was found to have a length of 2.25 inches and a width of 2.05 inches. The total axial error is four times the mean feature length and the total circumferential error is equal to the average feature width. The standard deviation of the feature sizing parameters was found using the following formula.

Formula 2: Standard Deviation = $\sqrt{\frac{\sum(x-\bar{x})^2}{(n-1)}}$

The standard deviation of feature length is 2.19 inches and the standard deviation of feature width is 1.01 inches. In both the circumferential and axial cases the total error is at least twice the standard deviation. It is the purpose of this statistical analysis to conclude that the total error is very significant compared to feature size. Considering the comparisons made, it is reasonable to proceed assuming this conclusion is valid.
Next, the effect of this error on a corrosion metal loss box plotted in the 2-D wrapping plane will be considered. In the plot shown below, the previously described corrosion boxing process is applied to a corrosion metal loss feature with average size parameters. The second corrosion rectangle is the same exact size, but is shifted by the amount of the total error. The total error shift is a 9 inch increase in axial location and a 2 inch or 20 minute increase in the O'clock position.

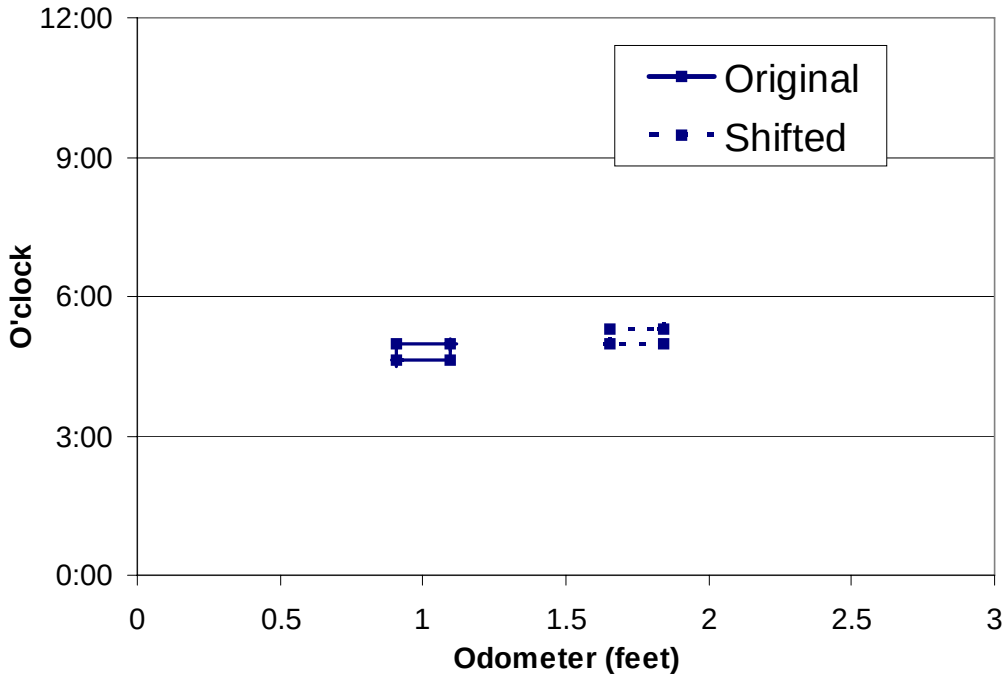


Figure 3-16: Average Size Corrosion Box and a Duplicate Shifted by the Magnitude of the Total Error

It is apparent from Figure 3-16 that the shifted corrosion rectangle does not overlap its original location counterpart at all. This presents a serious problem when an attempt is made to match this corrosion rectangle to a corresponding rectangle from an historically different ILI data set. The algorithm only assumes that two corrosion rectangles are matching if they are overlapping in the 2-D wrapping plane. If two corrosion rectangles from different historical data sets are both measured from the same physical corrosion than they should be identified as matching by the algorithm. However, if the corrosion rectangles are around average size and one is shifted by a magnitude equal to the maximum ILI error, then the algorithm might not detect these corrosion rectangles to be matching. This issue has been addressed in the program by way of a user defined matching tolerance. Different matching tolerance values can be assigned for both the axial and circumferential directions. Matching tolerances allow the program to match boxes even if they are not directly overlapping. The program will consider the corrosion boxes to be matching if they are within the specified tolerance distance from one another.

In this way, the user can decide if they would like to use the instrument's total errors as tolerance values, or they could use other more appropriate tolerances if they see fit.

3.3 Clustering

An important feature that has been written into the corrosion box matching program is called clustering. Clustering is a process applied to corrosion metal loss data acquired from a single in-line instrument run. The process of clustering can be described as grouping multiple corrosion metal loss boxes within a certain proximity to one another into one larger box. This larger box or cluster is exactly sized to enclose all of the clustered original corrosion metal loss boxes. This proximity value is similar conceptually to the matching tolerance discussed previously. It consists of user defined values in the axial and circumferential directions that dictate the maximum distance that can exist between two corrosion metal loss boxes that should be clustered together. Below Figure 3-17 shows three corrosion metal loss boxes overlapping one another. Figure 3-18 shows how these boxes would be clustered into one larger box.

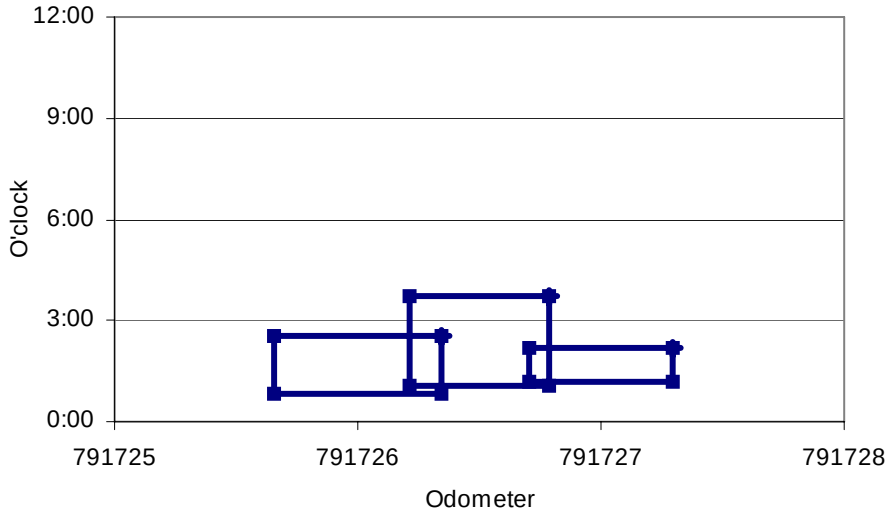


Figure 3-17: Corrosion Metal Loss Boxes that Should Be Clustered

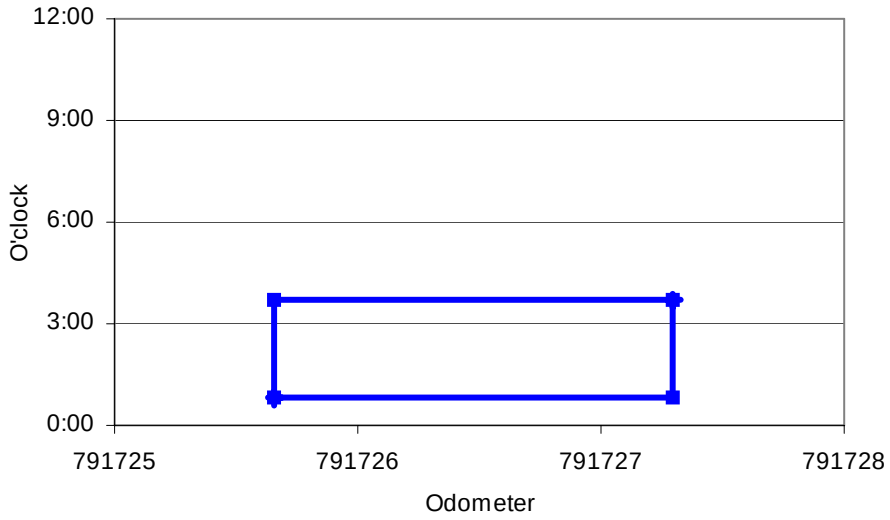


Figure 3-18: Cluster Box of Three Corrosion Boxes from Figure 3-17

Now that the process of clustering has been explained, it is necessary to explain the justification for this process. When observing Figure 3-18 above one might come up with a reasonable objection to clustering. If the large cluster box is being associated with corrosion in a similar manner that the original corrosion metal loss boxes were associated with corrosion, then we are associating some areas with corrosion where none has been detected. These areas are highlighted in Figure 3-19.

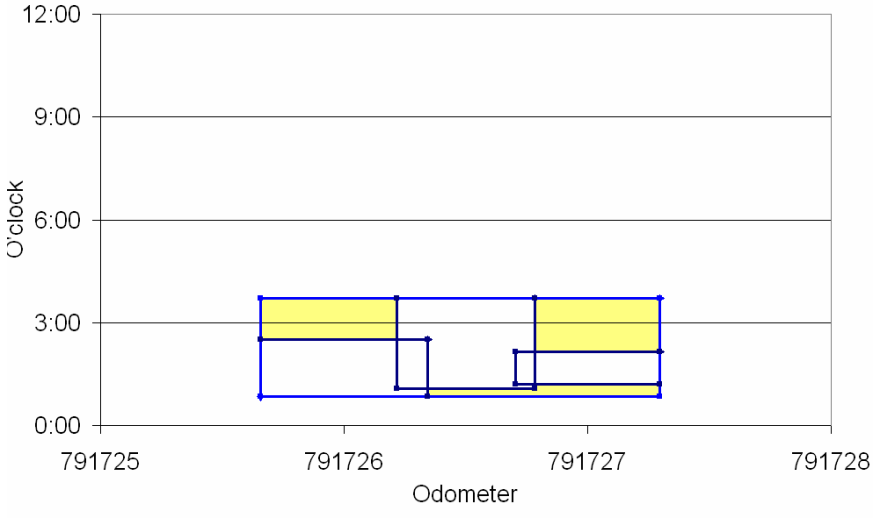


Figure 3-19: Cluster of Corrosion Boxes; Non-Corrosion Areas are Tinted Yellow

The reason that this is justified is the significance of the error associated with in-line instruments. The total axial and circumferential error is very significant compared to the proximal distances between reported corrosion metal loss features. The same ILI sample of 5,347 corrosion metal loss features used previously was used to analyze this assumption. It was found that 50.8% of the reported corrosion metal loss features were predicted to be within one total error distance from the next nearest corrosion metal loss feature or cluster in either the axial or circumferential directions. To illustrate the problems that this would cause when trying to match historical ILI corrosion data, the Figure 3-20 superimposes corrosion boxes from two historically different ILI runs on one plot. All of the corrosion boxes are close to the average size. Corrosion metal loss from the year 2000 is shown as red boxes, while corrosion metal loss from the year 2003 is shown with blue boxes. The error bars extending from the sides of each box are scaled to represent the total error in the axial and circumferential directions, respectively.

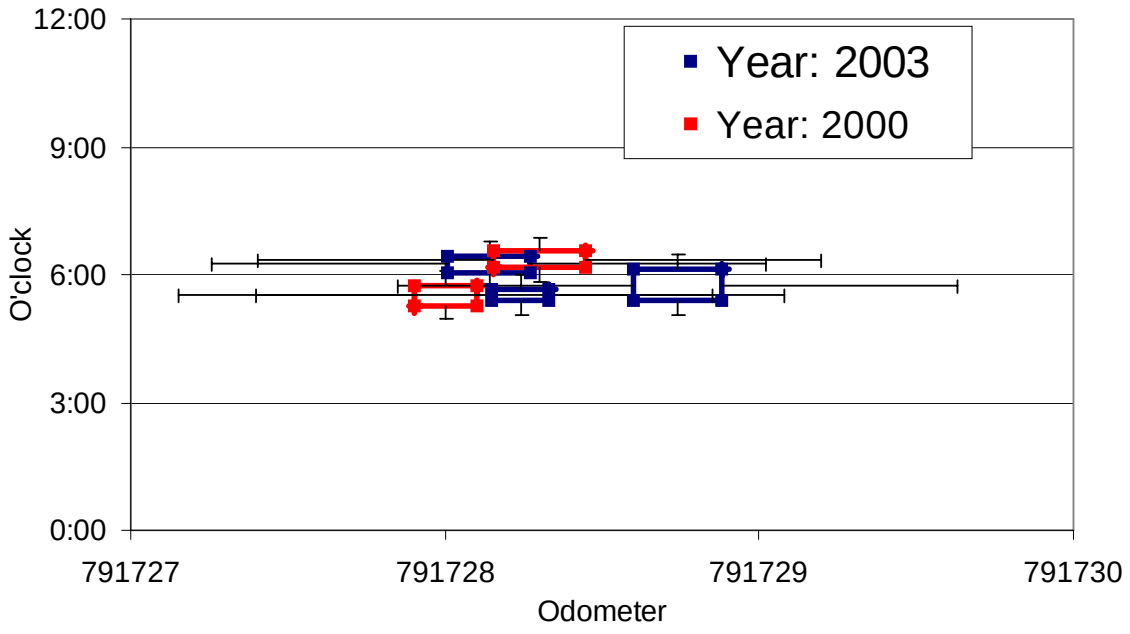


Figure 3-20: Unclustered Matching Boxes with Error Bars

When observing this figure it is apparent that the error bars dominate any meaningful matching information that might be conveyed. Because the error bars are so significant it is not justifiable to match any of the features shown. This situation is common in actual in-line instrument data. It is a situation when clustering can be a very useful tool. When the three corrosion metal loss boxes from the year 2003 are clustered in blue and the two corrosion boxes from 2000 are clustered red, the match is much more convincing. Using this technique the three measured features from 2003 can be simultaneously compared to both of the features from the year 2000. The resulting clusters are plotted in Figure 3-21 to show how matching the data becomes simpler.

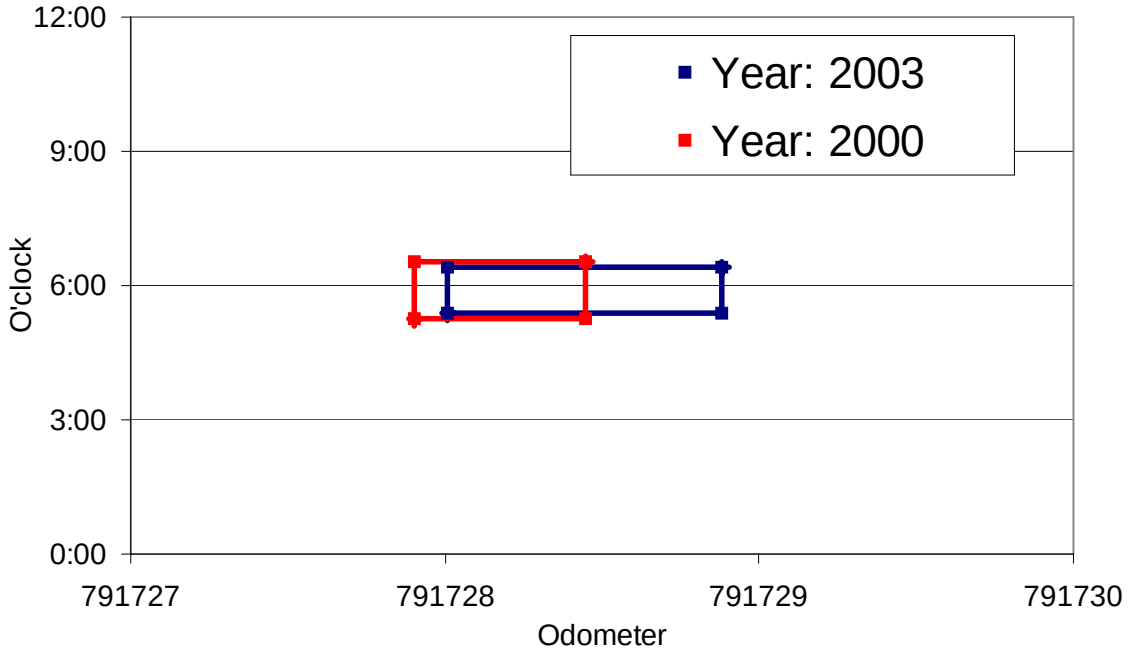


Figure 3-21: Matching Clusters of Corrosion Boxes from Figure 3-20

Even though the clusters do not directly represent the measured corrosion metal loss, they are a useful tool to match closely grouped corrosion features. The measurements associated with these features can then be compared to quantify how they have changed and the techniques of the box matching algorithm remain intact and sound.

4. Conclusions

In conclusion, it is relevant to discuss the main parts of the approach in terms of their relative degree of confidence. The first main part of the approach was identifying box overlap in a wrapping 2-D plane, which is representative of a pipe surface. There is a high degree of confidence in the algorithm to determine box overlap. It has been tested rigorously and is believed to have the capabilities to determine all instances of box overlap in a 2-D wrapping plane. There is also a high degree of confidence in the ability of the program based on this algorithm to detect box overlap quickly and accurately. The program has been tested on two samples of in-line instrument data collected in the field from a segment of pipe 44 miles long, which contain a total of over 9,000 instances of reported corrosion metal loss. The algorithm has the capability to identify all cases of box overlap within these samples in less than five minutes. The program is a success in the terms that it operates accurately and efficiently on large sets of in-line instrument data.
The second main part of the approach was corrosion matching. Recall, matching implies that when two reported corrosion metal loss boxes occupy the same area on the 2-D wrapping plane, they are assumed to represent the same physical corrosion on the pipe. This statement has been made from reasonable engineering deduction, but must be researched further before the actual confidence in matching can be quantified. The first tool I would recommend that the next researcher use would be the quantified in-line instrument error for the data being analyzed. Since this error has been found with an associated confidence level it might be possible to use these previous findings to determine a confidence level in corrosion matching. The second tool that I would recommend would be laboratory experiment and field validation. If the next researcher had the capability to verify that the physical corrosion metal loss was lining up with the findings from the matching technique, then it would be possible to quantify the corrosion matching error in this way as well.
During this research there have been some significant steps made in identifying all cases of box overlap in a wrapping 2-D geometry. However, until the errors associated with corrosion matching can be conclusively quantified, this program should still be considered experimental.

5. Acknowledgements

I would like to thank those who helped in the completion of this research and report. This includes the employees of CC Technologies Technologies for assisting me with data processing and analysis issues. I would also like to thank Dr. Tom Bubenik, my industrial advisor, for lending his expertise in the field of pipeline integrity. Most of all, I would like to thank Professor Marcelo Dapino from The Ohio State University for his mentorship and advice through my entire research process.

6. References:

1.American Petroleum Institute Website. About Oil and Natural Gas Page. http://www.api.org/aboutoilgas/

2. Baker Hughes Website. Pipeline Services Page. http://www.bakerhughesdirect.com/cgi-bin/bpc/resources/ExternalFileHandler.jsp?BV_SessionID=@@@@0770230978.1172377844@@@@&BV_EngineID=ccccaddkfgjfmmlcefecfefdfmldhfk.0&path=private/BPC/public/pipeline_services/index.html&channelId=-4207386

3. BJ Services Website. Inspection Services Page. http://www.bjservices.com/website/pps.nsf/InspectionServicesFrameset?openframeset

4. Bubenik, T.A., J.B. Nestleroth, and B.N. Leis. Introduction to Smart Pigging in Natural Gas Pipelines. Technical Presentation and Report. February 2001.

5. Clark, E.B., B.N. Leis, and R.J. Eber. Integrity Characteristics of Vintage Pipelines. Technical Report. October 2004.

6. GE Energy Website. Pipeline Integrity Management Page. http://www.gepower.com/prod_serv/serv/pipeline/en/index.htm

7. Reber, K., and M. Beller. Ultrasonic In-Line Inspection Tools to Inspect Older Pipelines for Cracks in Girth and Long-Seam Welds. Technical Report. 2003.

8. Thompson, Neil G., and Patrick H. Vieth, Corrosion Costs U.S. Transmission Pipelines as much as $8.6 Billion/Year. Pipeline & Gas Journal, March 2003.

9. Tuboscope Website. Products Page. http://www.tuboscope-pipeline.com/Products.htm

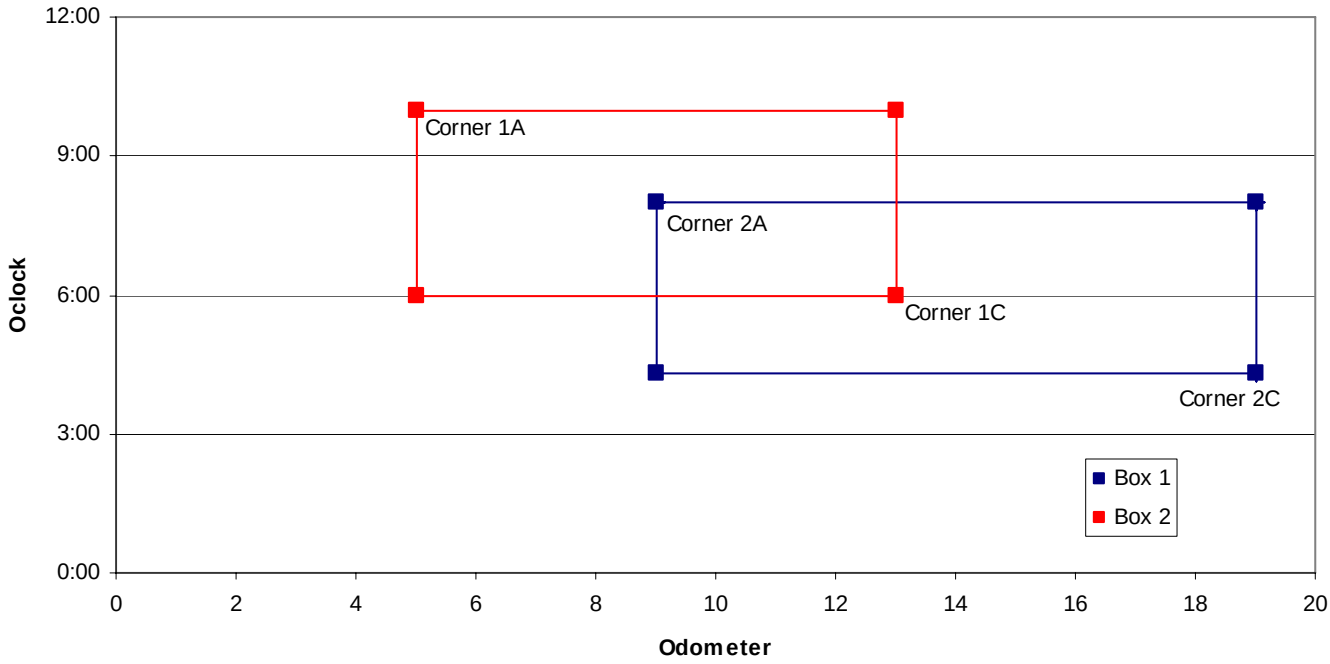
APPENDIX

Appendix A. Box Overlap Algorithm

Unique Overlap Cases with Associated Logic Operators

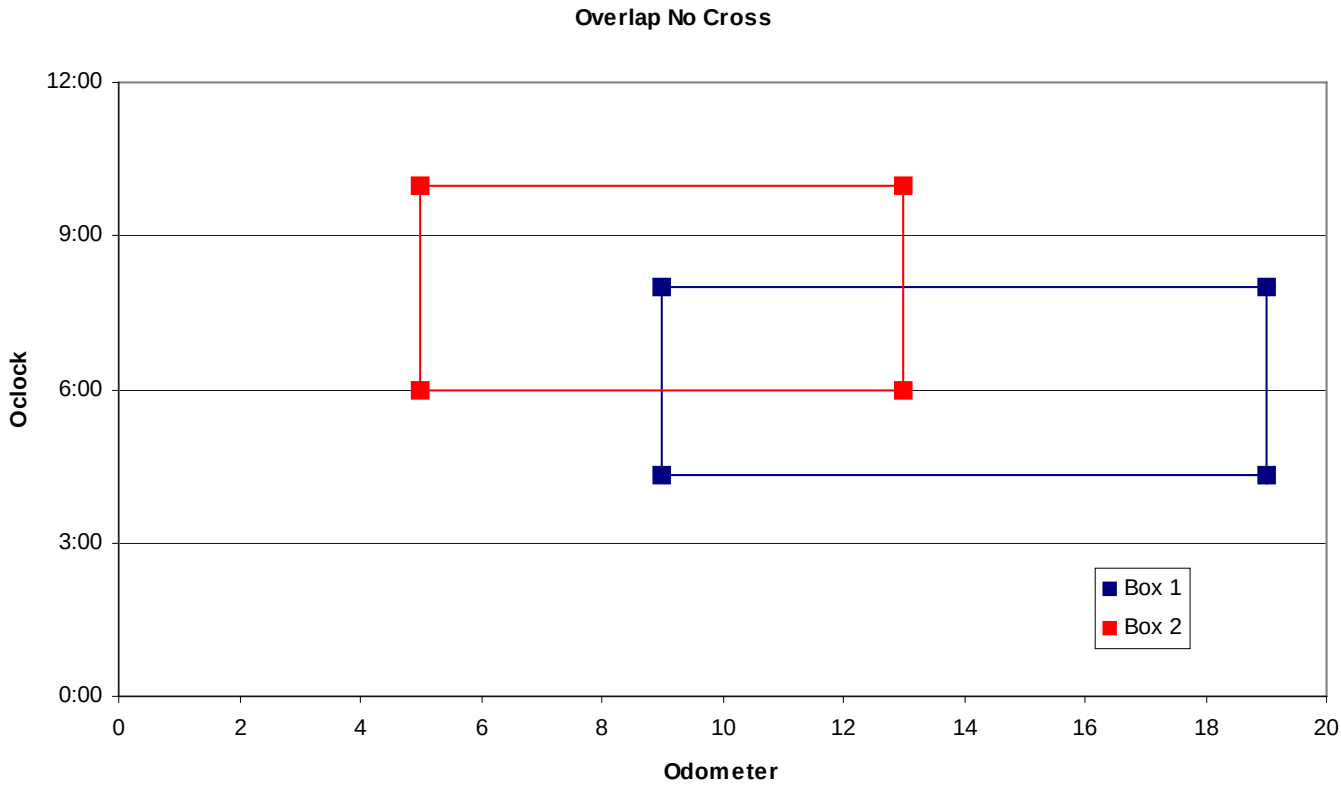
Explanation of Algorithm Logic Conditions

Conditions:
1Ax: The odometer value of corner A of Box 1
1Ay: The O'clock value of corner A of Box 1
1Cx: The odometer value of corner C of Box 1
1Cy: The O'clock value of corner C of Box 1
2Ax: The odometer value of corner A of Box 2
2Ay: The O'clock value of corner A of Box 2
2Cx: The odometer value of corner C of Box 2
2Cy: The O'clock value of corner C of Box 2
Cross (1): Equals 1 when Box 1 is crossing the 12:00 orientation line, equals 0 otherwise
Cross (2): Equals 1 when Box 2 is crossing the 12:00 orientation line, equals 0 otherwise
Cross360 (1): Equals 1 when Box 1 is a 360 degree Box, equals 0 otherwise
Cross360 (2): Equals 1 when Box 2 is a 360 degree Box, equals 0 otherwise



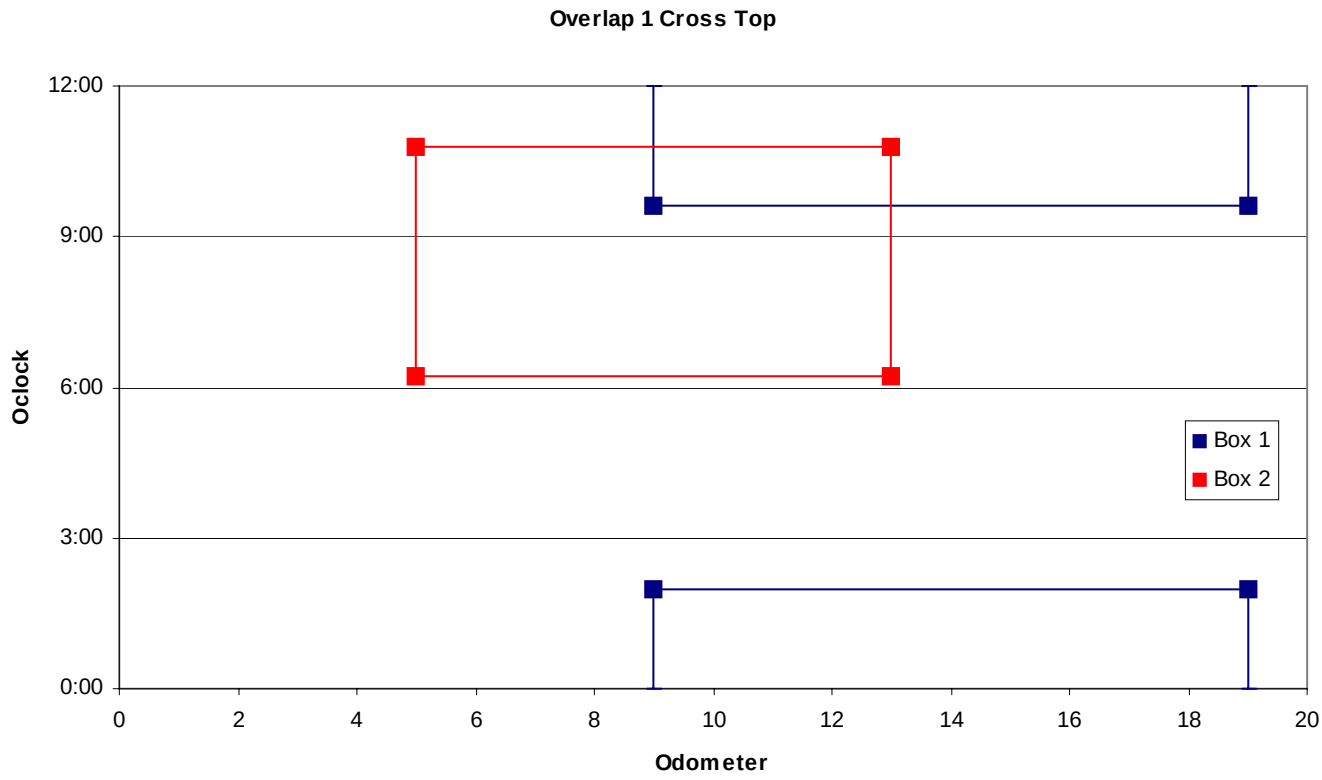
Case 1: Overlapping boxes; both are not crossing 12:00

Conditions:
1Ay>2Cy
1Ax<2Cx
1Cx>2Ax
1Cy<2Ay
Cross (1) = 0 (off)
Cross (2) = 0 (off)
Cross360 (1) = 0 (off)
Cross360 (2) = 0 (off)



Case 2: Overlapping boxes with box 1 crossing 12:00. Overlap is near the top of the 2-D wrapping plane

Conditions:
1Ay < 2Cy
1Ax < 2Cx
1Cx > 2Ax
1Cy < 2Ay
cross (1) = 1 (on)
cross (2) = 0 (off)
cross360 (1) = 0 (off)
cross360 (2) = 0 (off)

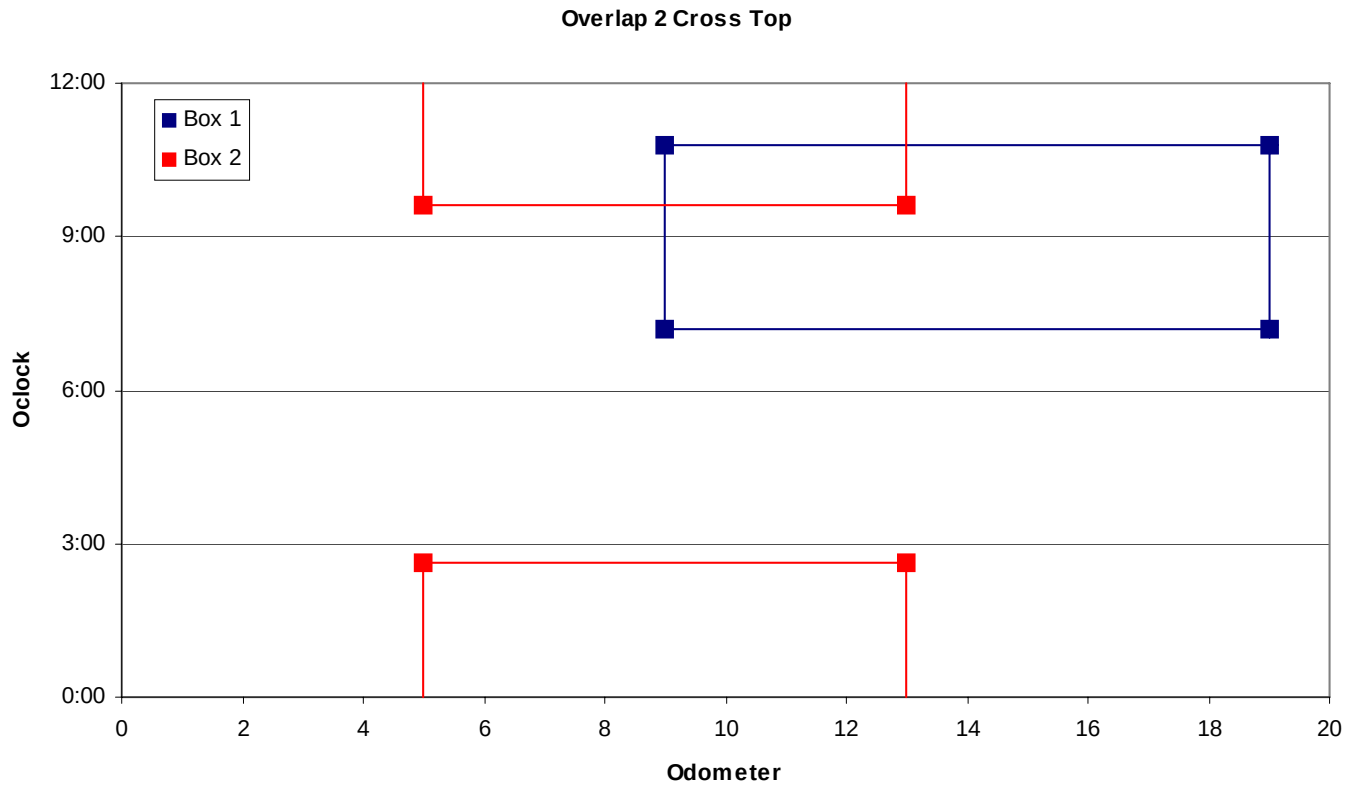


Case 3: Overlapping boxes with box 1 crossing 12:00. Overlap is near the bottom of the 2-D wrapping plane

Conditions:
1Ay > 2Cy
1Ax < 2Cx
1Cx > 2Ax
1Cy > 2Ay
cross (1) = 1 (on)
cross (2) = 0 (off)
cross360 (1) = 0 (off)
cross360 (2) = 0 (off)

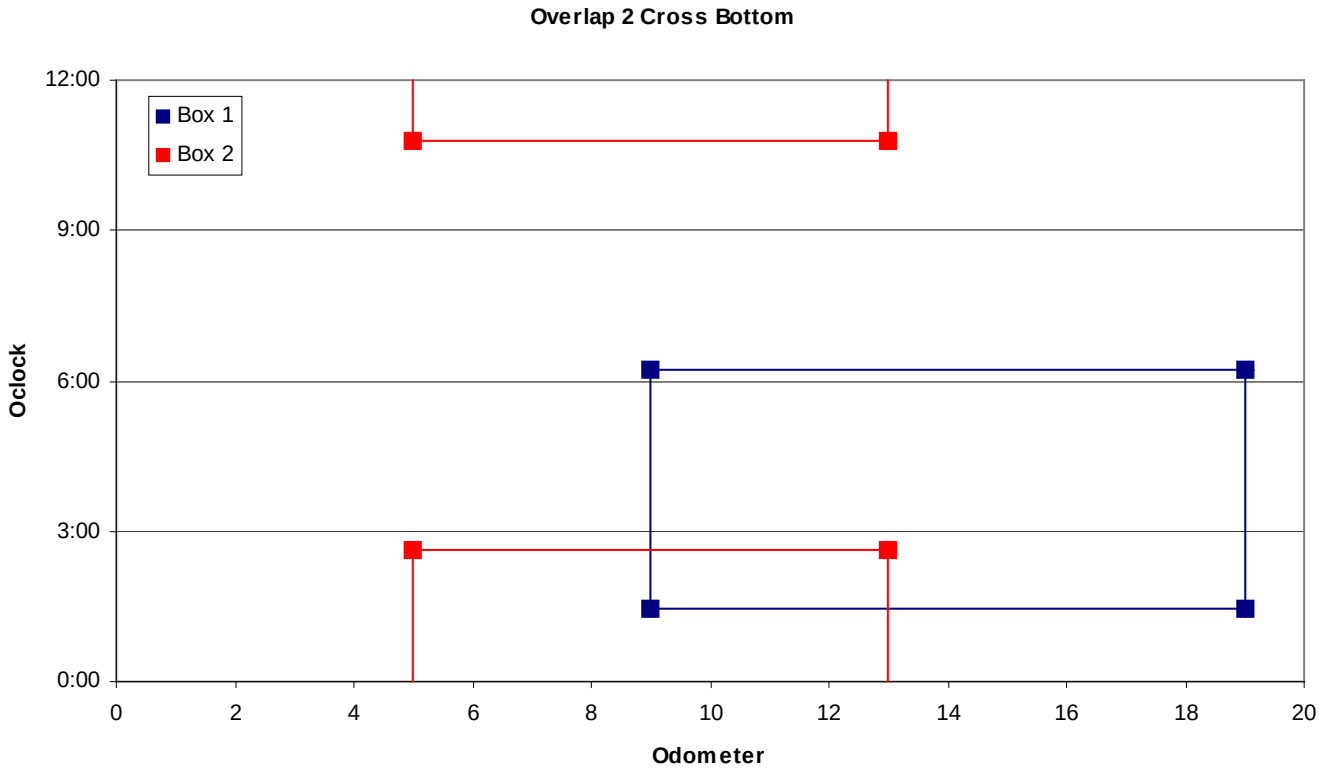
Case 4: Overlapping boxes with box 2 crossing 12:00. Overlap is near the top of the 2-D wrapping plane

Conditions:
1Ay > 2Cy
1Ax < 2Cx
1Cx > 2Ax
1Cy > 2Ay
cross (1) = 0 (off)
cross (2) = 1 (on)
cross360 (1) = 0 (off)
cross360 (2) = 0 (off)



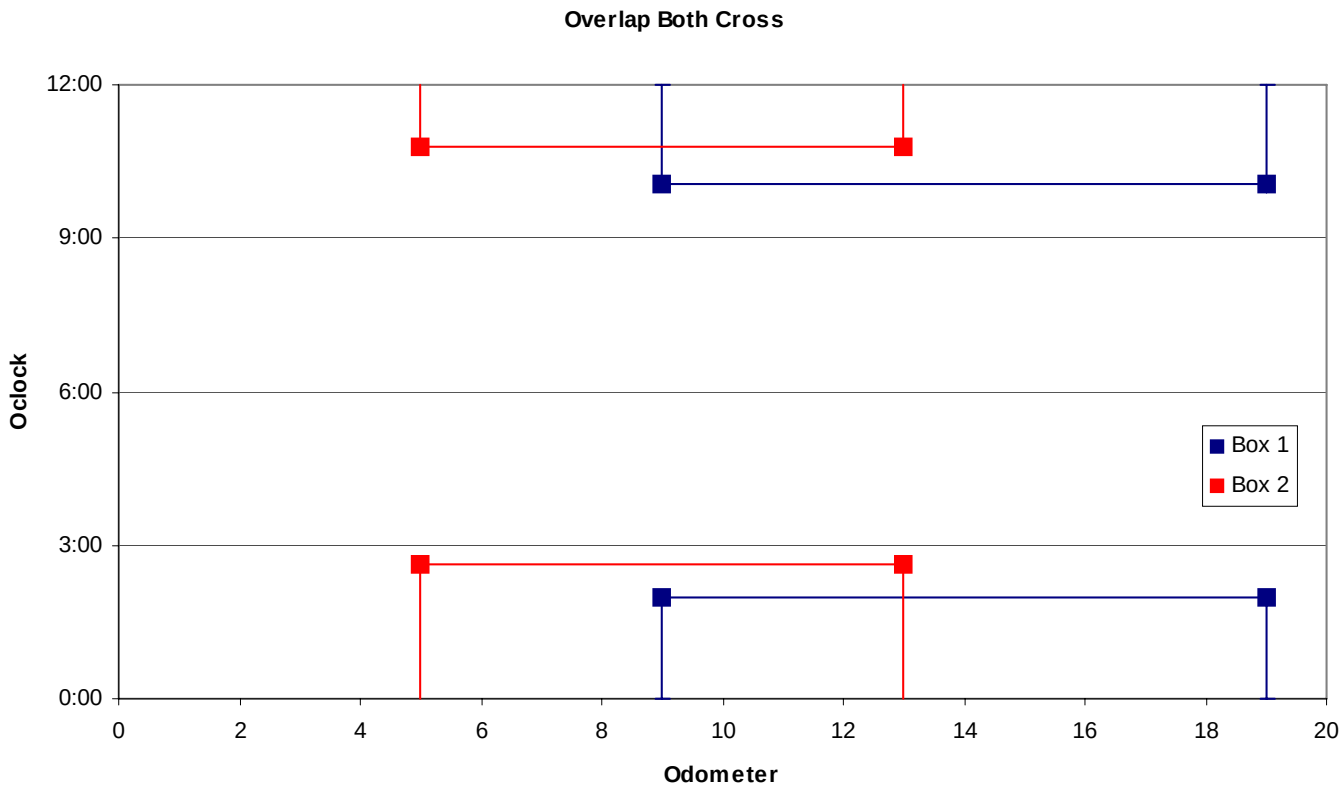
Case 5: Overlapping boxes with box 2 crossing 12:00. Overlap is near the bottom of the 2-D wrapping plane

Conditions:
1Ay < 2Cy
1Ax < 2Cx
1Cx > 2Ax
1Cy < 2Ay
cross (1) = 0 (off)
cross (2) = 1 (on)
cross360 (1) = 0 (off)
cross360 (2) = 0 (off)



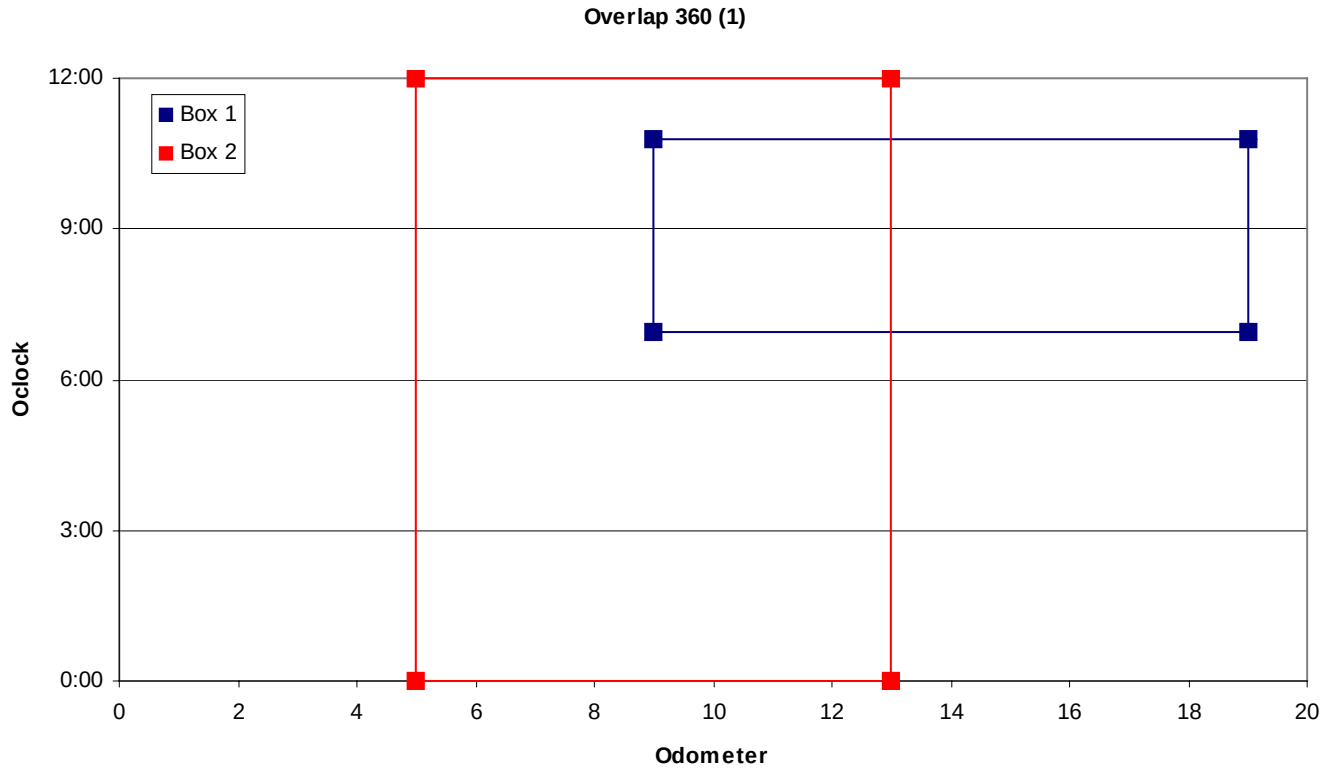
Case 6: Overlapping boxes with both boxes crossing 12:00.

Conditions
1Ax < 2Cx
1Cx > 2Ax
cross (1) = 1 (on)
cross (2) = 1 (on)
cross360 (1) = 0 (off)
cross360 (2) = 0 (off)



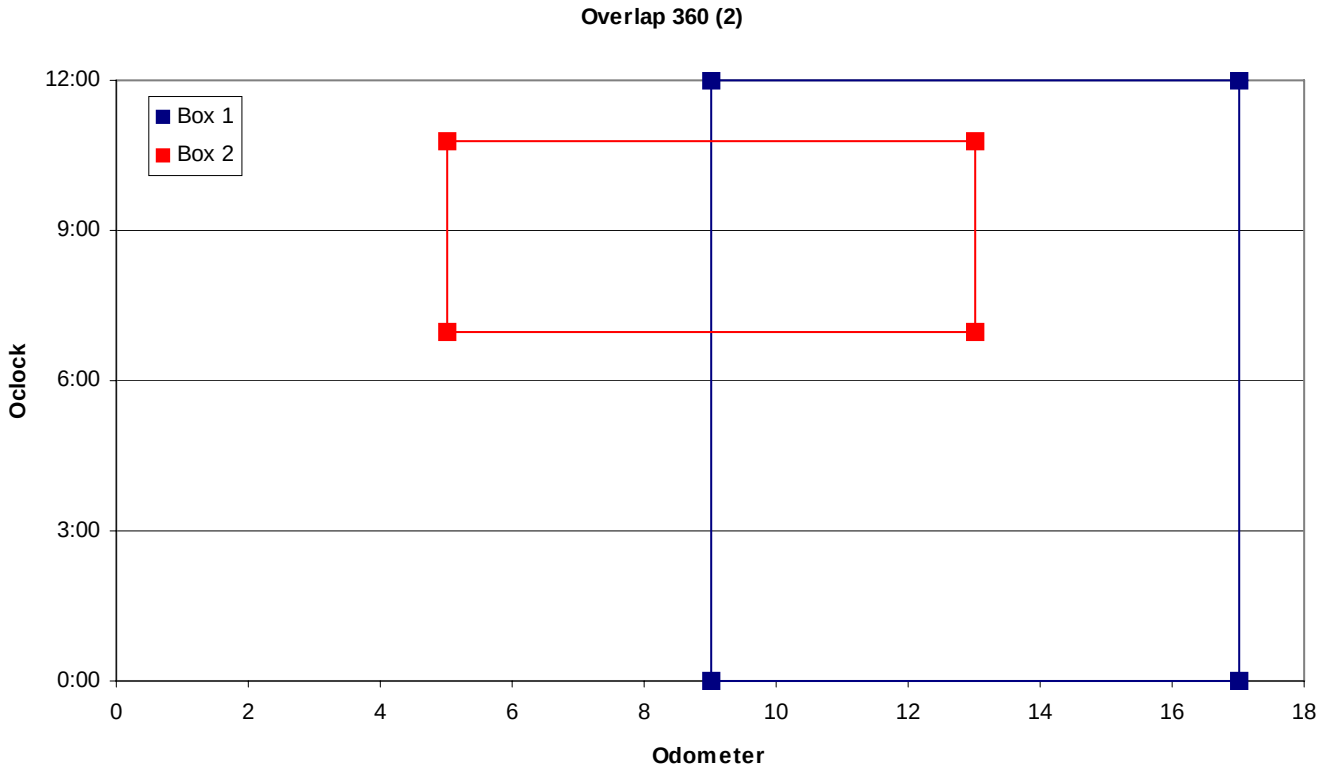
Case 7: Overlapping boxes when box 2 is a 360 degree box.

Conditions:
1Ay > or < 2Cy
1Ax < 2Cx
1Cx > 2Ax
1Cy < or > 2Ay
cross (1) = 0 or 1
cross (2) = 0
cross360 (2) = 1
cross360 (1) = 0



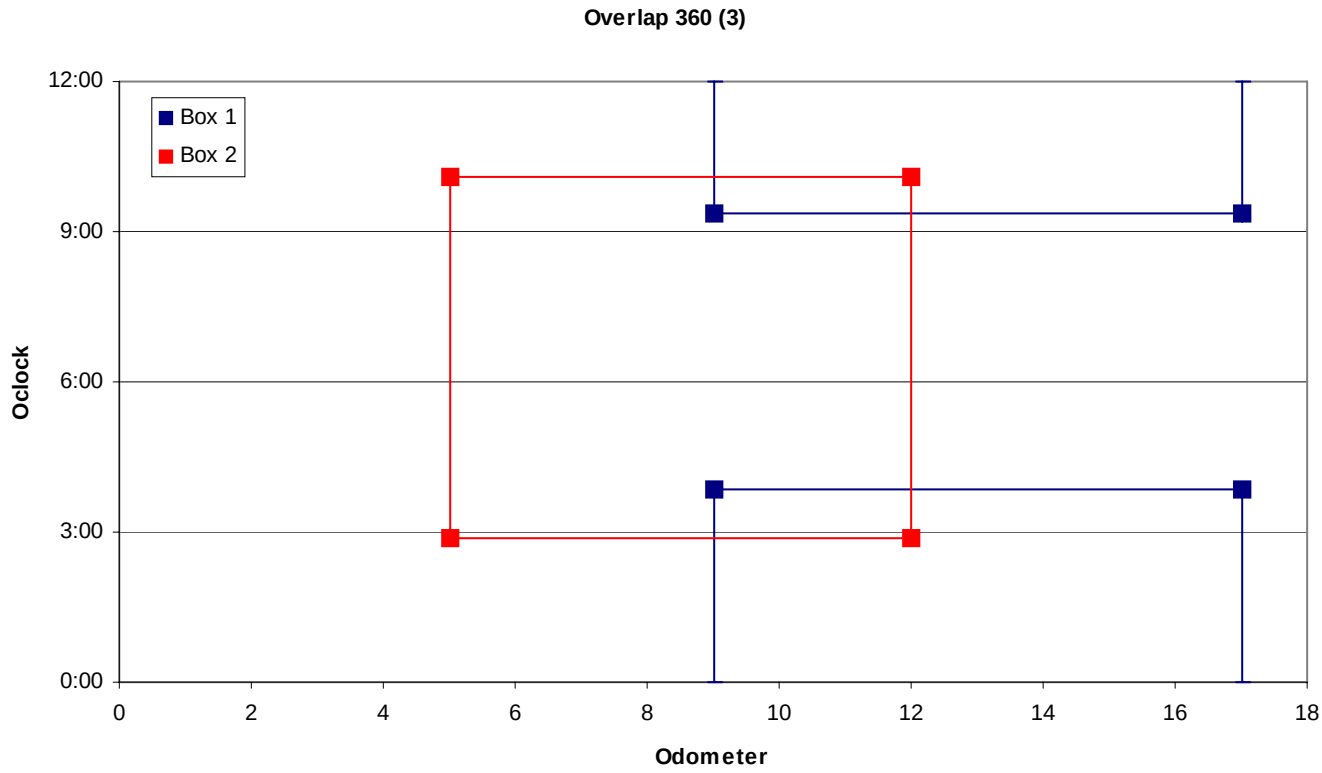
Case 8: Overlapping boxes when box 1 is a 360 degree box.

Conditions:
1Ay > or < 2Cy
1Ax < 2Cx
1Cx > 2Ax
1Cy < or > 2Ay
cross (1) = 0
cross (2) = 0 or 1
cross360 (2) = 0
cross360 (1) = 1



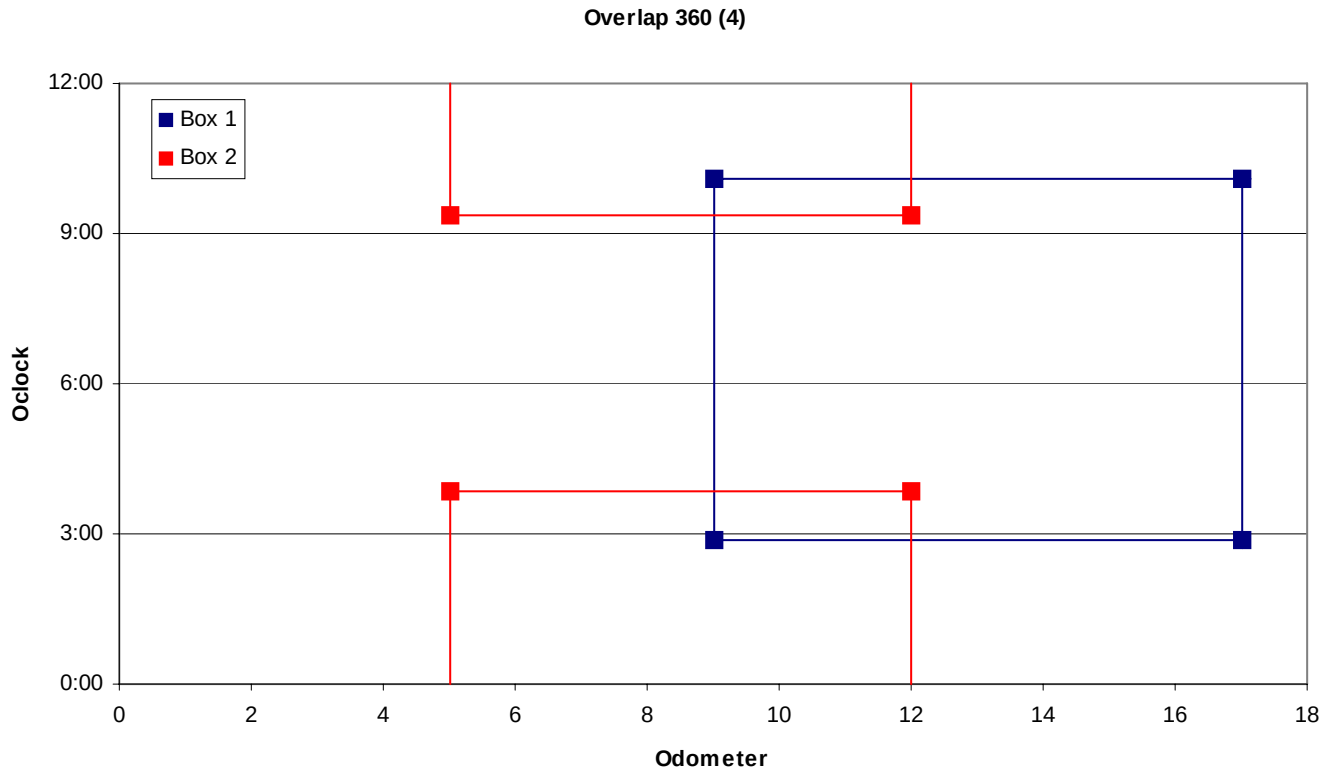
Case 9: Overlapping boxes when box 1 is crossing 12:00 and box 2 is not.

Conditions:
1Ay > 2Cy
1Ax < 2Cx
1Cx > 2Ax
1Cy < 2Ay
cross (1) = 1
cross (2) = 0
cross360 (2) = 0
cross360 (1) = 0



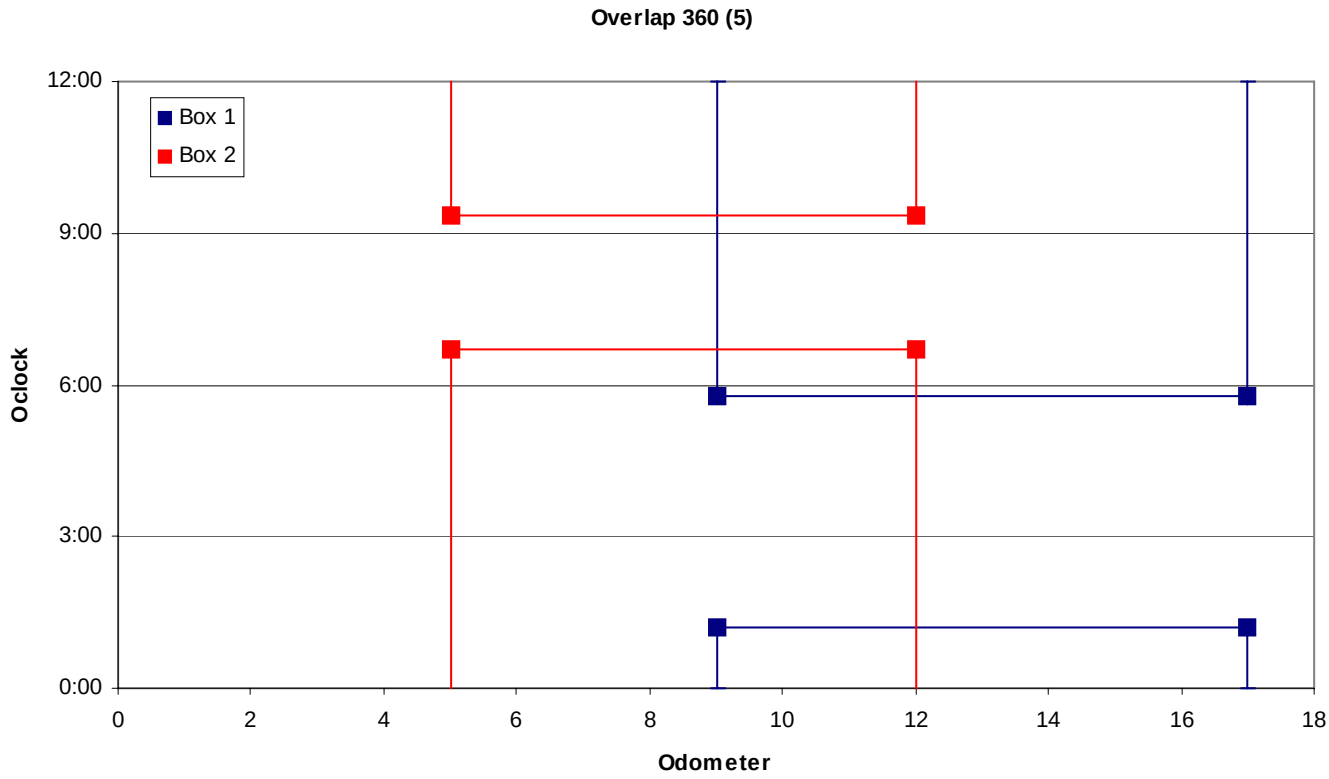
Case 10: Overlapping boxes when box 2 is crossing 12:00 and box 1 is not.

Conditions:
1Ay > 2Cy
1Ax < 2Cx
1Cx > 2Ax
1Cy < 2Ay
cross (1) = 0
cross (2) = 1
cross360 (2) = 0
cross360 (1) = 0



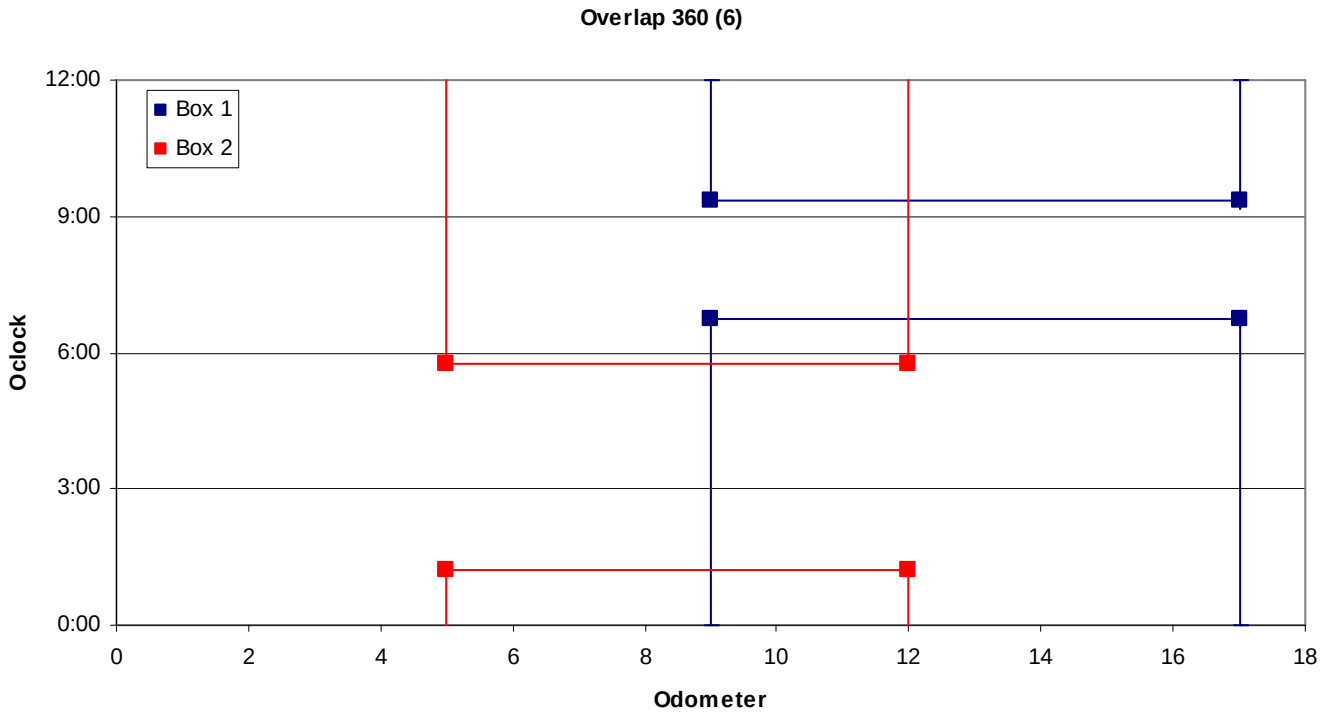
Case 11: Overlapping boxes when both boxes are crossing 12:00 (a).

Conditions:
1Ax < 2Cx
1Cx > 2Ax
1Cy < 2Ay
cross (1) = 1
cross (2) = 1
cross360 (2) = 0
cross360 (1) = 0



Case 12: Overlapping boxes when both boxes are crossing 12:00 (b).

Conditions:
1Ax < 2Cx
1Cx > 2Ax
1Ay > 2Cy
cross (1) = 1
cross (2) = 1
cross360 (2) = 0
cross360 (1) = 0



Appendix B. Read Me Instruction File for the Box Matching Program

Read Me:

This program allows the user to input in-line instrument (ILI) metal-loss corrosion data from two tool runs taken in the same pipe at different times. The input data must include location information including axial location in feet and circumferential location as an O'Clock. Size information must be input as well for each metal-loss corrosion feature, including length and width in inches and depth as a percentage of wall thickness. The pipe diameter must also be input by the user.
This program clusters metal-loss corrosion features from the same run together if they are within a clustering tolerance which is input by the user for each run. The program uses a matching tolerance which is input by the user to match clusters from subsequent runs. The program detects which clusters are present, within this tolerance, to the same location in both runs.
The inputs to the program and the outputs generated by the program are placed into an Excel workbook. The program is a series of scripts run in MatLab.

Instructions:
1. Open the Excel File titled "Box_Matching_Input-Output_File."

2. Input ILI metal-loss corrosion data into Columns F through L starting with row 2 in the 'input1' and 'input2' sheets in the "Box_Matching_Input-Output_File." A short definition for each data type is given below:
Reference Number; Unique number assigned by the In-Line Instrument to every detected feature including metal-loss corrosion
Pipeline Feature; Text Name (they should all be metal-loss corrosion features)
Odometer; Axial location of feature in feet
Predicted Depth; Feature depth as a percentage of wall thickness
Predicted Length; Feature length in the axial direction in inches
Predicted Width; Feature width in the circumferential direction in inches
O'Clock Orientation; Circumferential location of feature in O'Clock
(Note: That Sample data may be present in the input columns.)

3. Enter Pipe Diameter in inches in cell D3 of the 'input1' sheet.

4. Enter clustering tolerance values. Clustering tolerances dictate how proximal two features within the same ILI tool run must be to one another to be clustered together. These include axial and circumferential tolerances in inches. They should be entered in cells D6 and D7 in both 'input1' and 'input2' sheets.

5. Enter matching tolerance values. Matching tolerances dictate how proximal two clusters from sequential ILI tool runs must be to one another to be matched together. These include axial and circumferential tolerances in inches. They should be entered in cells D10 and D11 in the 'input1' sheet.

6. Save the Excel file “Box_Matching_Input-Output_File.”

7. Open MatLab.

8. Change the directory to the “Thesis Batch Folder.”

9. Run the M-file titled “Start”

10. The program will probably take 3-5 minutes to complete. When the program is “busy,” data is being imported from or exported to Excel. When the program isn’t “busy,” a progress meter should be displayed to show the user that the program is working.

11. When the program is finished the user should bring the Excel file to the top of the desktop. The output sheets are now full. The output plot(s) is/are now full.

A description of each output sheet/plot is given below:

ClusterList1: This sheet gives a complete list of all clusters formed from the input1 data set. The list includes a unique number assigned by the program to each cluster (column A), the number of ILI identified metal-loss corrosion features within each cluster (column B), and a list of the unique reference numbers for all ILI identified metal-loss corrosion features within each cluster (beginning at column N). The list also includes the locations of each cluster corner point in terms of its axial location in feet and its circumferential location in O’Clock.

ClusterList2: This sheet is similar to ClusterList1, except that this sheet is in reference to the input2 data.

OrphanList1: This sheet gives a list of all clusters formed from the input1 data set that did not match any clusters from the input2 data set.
The list includes:
Column A: A unique number assigned by the program to each cluster
Column B: The number of ILI identified metal-loss corrosion features within each cluster
Column C: The summed area of all ILI identified metal-loss corrosion features within each cluster (sqin.)
Column D: The area enclosed by the cluster dimensions (sqin.)
Column E: The maximum depth for all features within this cluster (% of wall thickness)
Column F: The minimum depth for all features within this cluster (% of wall thickness)
Columns G-N: The coordinates of the cluster’s corner points in axial feet and circumferential O’Clock
Column R: The length of the cluster in feet (appropriate units for plotting)
Column S: The width of the cluster in O’Clock time (appropriate units for plotting)

OrphanList2: This sheet is similar to OrphanList1, except that this sheet is in reference to the input2 data.

OneToOneList: This sheet gives a list of clusters that matched exactly one to one between subsequent tool runs. The information given is the same as that given in the OrphanList1 sheet. The data is given for both matching clusters in the same row. The data from the input1 data set is in columns A-S and the data from the input2 data set is in columns U-AM.

OneToMultiList1: This sheet gives a list of clusters from the input1 data set that either matched more than one cluster from the input2 data set or matched one cluster from the input2 data set that in turn matched more than one cluster from the input1 set. None of these matches are one to one. The information is similar to the OneToOneList, but instead of listing the matching cluster in the same row the matching cluster list occurs in the subsequent rows. The data from the input1 data set is in columns A-S and the data from the input2 data set is in columns U-AM. The zeros represent blank spaces.

OneToMultiList2: This sheet is similar to the OneToMultiList1, except it lists the clusters from the input2 data set in columns A-S and the data from the input1 set in columns U-AM.

MatchPlot: This plot shows all the clusters from the input1 and input2 data sets as rectangles. The rectangles from the input1 clusters have orange corners, while the rectangles from input2 have blue corners. The rectangles have different colors according to their match type. The orphans have been plotted as red rectangles. The orphans all appear individually with no other rectangles within the tolerance range. The one-to-one matches were plotted as green rectangles. The one to one matches always appear in pairs, one from the input1 set and one from the input2 set. The one to multi matches were plotted as yellow rectangles. These matches always appear with more than two others within the matching tolerance range. (Note: The data ranges within the plot may have to be adjusted accordingly to each data set)