SUPPLEMENTAL MATERIALS

ASCE *Journal of Water Resources Planning and Management*

Conceptual Water Main Failure Risk: Self-Excitation, Pipe Age, and Statistical Modeling Performance

Charles Hammond, Jiancang Zhuang, Casey LeBlanc, Sarah Rahimi-Ardabily, Tao Zhang, Robert Good, and Frank Loge

DOI: 10.1061/JWRMD5.WRENG-6432

Contents

This supplement contains additional analysis of the results and additional information regarding several key methods used in this study: 1) the algorithm used to infer missing pipes and to assign mains to breaks, 2) the model parameter estimation method, and 3) the simulation algorithm. The supplemental figures are found at the end of this document. Some general comments: Python was used as the main programming language, though computational limits imposed by the estimation algorithm require the use of something like Cython to speed up parts of the code. We recommend using Cython only for parts of the calculations that bog down your algorithm, for example, loops. Without reaching C-level speeds via the use of Cython or an equivalent, the estimation and simulation algorithms would take an intolerably long time to run.

1. Seasonality of Main Breaks

Many different factors contribute to main break rates (see Table S8). As shown in Figures S4, S5, and S6, the striking patterns observed for some types of main breaks suggests that there are underlying seasonal factors. The clearest and most significant patterns are found in circumferential and longitudinal CI breaks, and circumferential AC breaks, followed by CI blowouts and joint failures. The other types of AC breaks do not have significant seasonal patterns. Interestingly, CI and AC circumferential breaks have opposite seasonal behaviors: they peak in the winter (Dec/Jan) for CI and in the late summer/fall (Aug/Sep/Oct) for AC. Longitudinal CI breaks peak in summer (Jul/Aug), while longitudinal AC breaks do not appear to be seasonal. CI joint failures and blowouts have peaks in the summer (July and Jul/Aug/Sep, respectively), while blowouts also have a smaller peak in the winter (January). The lack of seasonality in ST main breaks is consistent with the literature (Barton et al., 2019). ST mains are generally newer and have not been exposed to environmental and operational stresses for nearly as long as many of the CI and AC mains, thus, as ST mains age and deteriorate we may eventually see seasonality in their break patterns as they become susceptible to environmental and operational stressors. To explain the patterns evident in CI and AC breaks, it is important to understand both the pipe materials and the mechanisms that lead to each failure mode.

CI pipes were produced and installed from the late 19th century to the late 1960s, when ductile iron became the primary iron water pipe material. They were produced by pit-casting until the 1920s, after which centrifugal casting became the predominant manufacturing method (Seica and Packer, 2004). CI consists of graphite flakes suspended in a matrix of iron; the microstructure produced by this combination causes CI to be more brittle than most metals (Seica and Packer, 2004). The graphite flakes concentrate

stress and initiate cracks when CI is placed under tension, and transmit stress when compressed. As a result, the compression strength of CI can be up to twice to four times the ultimate tensile strength (Seica and Packer, 2004). Data compiled by Seica and Packer (2004) show that pit-cast and centrifugal spun CI have different properties and that the properties vary significantly within each type, indicating that changing manufacturing practices have resulted in CI pipes with very different material properties: tensile strengths range from 33 to 305 MPa and compressive strengths range from 519 to 1047 MPa. CI pipes are considered rigid (Barton et al., 2019).

AC pipes consist of Portland cement, asbestos fibers, and sometimes silica, and were first installed in the 1930s (Task Committee on Water Pipeline Condition Assessment, 2017). Type I AC pipes have 80% cement and 20% asbestos, while Type II AC pipes have 40% cement, 40% silica, and 20% asbestos, rendering them more resistant to acids and sulfates (Task Committee on Water Pipeline Condition Assessment, 2017). Type II pipes were the dominant variety from 1940 onward, so the AC mains considered here are likely mostly Type II, as most of EBMUD's AC mains were installed after 1940. AC pipes are considered rigid and susceptible to ground movement (Barton et al., 2019).

The failure mode of a given break indicates the types of forces that were likely acting on the pipe at the time of the break. It is well-known in the literature that circumferential breaks are primarily caused by longitudinal stresses (thermal contraction, beam action, inferior installation practices, and third-party accidents) and longitudinal breaks are caused by hoop stresses (operating and surge pressure, soil cover load, traffic loads, and frost heave) (Rajani and Kleiner, 2001).

In the literature, CI breaks are reported to increase in frequency during the winter, which is consistent with the findings presented here indicating that circumferential CI breaks peak in December and January. The primary explanations for an increase in iron main breaks in the winter is frost heave and the associated soil expansion and cold internal water temperatures (Barton et al., 2019). However, the distribution network considered here experiences little to no frost, as it is located in a Mediterranean coastal climate. Furthermore, frost heave is associated with longitudinal rather than circumferential breaks. A more plausible explanation for the winter spike in circumferential CI main breaks is that cold reservoir water induces thermal stresses capable of causing circumferential breaks. In fact, Barton et al. (2019) cite research suggesting thermal stress from cold water may have a greater impact than frost. However, water treatment plant effluent temperature observations for EBMUD indicate that the lowest water temperatures experienced by pipes is approximately 55-60 °F in the months of December and January. Studies that mention thermal stress due to cold water usually refer to water temperatures near freezing. The relatively mild low temperatures experienced by the mains studied here may still contribute to main breaks through thermal stress, but other factors may be involved. Additional contributing factors might include 1) transfer of traffic loads during the rainy season when soil moisture increases unevenly underneath roads, leading to beam action, and 2) an unevenly distributed increase in soil weight due to increased moisture content. As soil becomes wetter, it generally becomes weaker (Pritchard et al., 2014). Despite the general belief that pavements are impermeable surfaces, Redfern et al. (2016) cites several studies providing evidence of low but significant infiltration of rainfall through deteriorated pavements.

Summer peaks in CI main blowouts, longitudinal breaks, and joint failures have not been reported in the literature; however, it may be that the more extreme winter conditions experienced by the distribution networks commonly studied overshadow any modest trends similar to those shown in Figure S5 (b, c, and d). Blowouts and longitudinal breaks are both driven by hoop stresses, which suggests that the driving factors may be related to pressure surges or vertical loading.

It is likely that the late summer/fall peak in AC circumferential breaks is caused by soil movements. The climate in which EBMUD operates is characterized by wet winters and dry summers. Furthermore, the local soils are diverse and sometimes have significant clay content. Maximum soil displacement/shrinkage, and thus maximum break volume, would be expected immediately before the first rains of the year, which is exactly what is shown in Figure S6 (a). This finding is consistent with the literature (Barton et al., 2019).

2. Inference

The fitted triggering parameters (α, γ) indicate that ST main breaks induce the largest but fastest decaying temporary increase in break risk, while CI has the second largest and second fastest decaying temporary increase in break risk, and AC has the lowest and slowest decaying increase in break risk (Table S7, Figure S12). This suggests that ST mains are at the greatest risk of incurring another break immediately following a break, followed by CI and AC, in that order. However, when considering the 95% confidence intervals for the parameter estimates, the immediate increase in risk for ST mains is indistinguishable from the increases in risk for CI and AC mains (see Figure S12). The temporary increase in risk due to a CI or ST break decays to negligible after approximately ten years, while it more than twenty years for AC, as shown in Figure S12. These differences may reflect the variability in the methods of repair for each material and/or the nature of the conditions that tend to lead to main breaks in different materials. While the estimates of the triggering parameters mostly agree with the empirical evidence shown for AC and ST in Figure 2, the theoretical decay for CI is faster than expected based on the data. Perhaps other covariates and factors become more important as the time since the last break increases beyond ten years for CI. These estimates provide evidence that the SSPP modeling error associated with omitted break history is likely insignificant for history more than ten to twenty years before the start of break records, depending the material.

The fitted model coefficients provide insight regarding the associations between covariates and main breaks. The CI model includes the largest number of covariates, with the most important factors, judged by the magnitude of the coefficients, being the length (positive coefficient) and diameter (negative coefficient) of the pipe, as shown in Table S7. These results are expected, as longer pipes present more opportunity to break and larger pipes resist bending stresses more effectively due to their larger area moment of inertia. Time varying and all-time average per pipe water demand have the next largest effects. The average per pipe water demand accounts for general local water demand conditions while the time varying water demand allows for the changes in demand over time to be assessed. Based on the coefficients, increased average per pipe water demand is associated with reduced break risk, meaning areas with higher average demand experience fewer breaks, while the change in demand over time is positively associated with break risk, meaning increases in demand over time for a given area correspond to higher break risk. This suggests that surges in demand might lead to increased break risk, while areas with generally higher demand experience reduced break risk.

CI main breaks also show a relationship with average ambient air temperature and precipitation, both of which have negative coefficients, suggesting that increased rainfall over a two-month period and increased temperatures are both associated with reduced CI break risk. These results appear to contradict the observation that CI breaks increase in both the winter and in the summer, when both precipitation and temperature are at their highest. However, water demand appears to be the dominant seasonal variable for CI breaks and thus the coefficients for other seasonal variables may not be intuitive when viewed alone. In general, the model for CI was unable to capture the observed seasonal fluctuation in breaks, suggesting that the model is inadequate for that purpose and is likely missing one or more important covariates. Thus, caution should be used when interpreting the coefficients in the context of seasonality. The bimodal

seasonal nature of CI breaks is likely difficult to model given the current approach. Future improvements of the SSPP model might include modeling the failure mode of the break, which would allow the model to reflect the strongly unimodal patterns underlying the aggregate seasonality observed in CI breaks. Future work might also include a more detailed investigation of the effects of traffic on CI main breaks, as one of the preliminary models found a significant but negative association. Correlation does not imply causation, and it is difficult to explain mechanistically why increased traffic might be related to reduced break risk, but possible explanations might include sturdier construction methods for mains experiencing heavy traffic or possibly that mains in areas with heavy traffic may be located away from direct exposure to traffic stress; however, average traffic volume was not significant for any of the models presented in Table S7.

Intuitively, the modelling results suggest that mains installed more recently are associated with decreased break risk, though this seemingly contradicts the small but negative coefficient for the age of the pipe. It also is found that the Euclidean distance from a pump station to a CI main is important, with mains further away from pump stations experiencing reduced break risk. Pump stations are known sources of water hammer, and it is intuitive that mains that are further away will be more protected from such sources of stress. It is also observed that gravity-fed CI mains have a slightly reduced break risk as compared to pump-fed CI mains. The clay content of the soil appears to influence CI breaks, with higher clay contents associated with higher break risk.

The model for AC main breaks indicates that ambient air temperature is the only important seasonal variable of those considered here. The increase in break risk as temperatures increase agrees with the summer peak in breaks shown in Figure S6. As temperatures increase and precipitation ceases, soils dry and shrink, often unevenly. Differential settlement of soils due to drying is a commonly cited cause of circumferential AC main breaks. The positive coefficient for both ambient air temperature and clay content suggest that differential settlement is indeed a driving factor for AC main breaks. The model also suggests that the soil pH is negatively associated with break risk, meaning more acidic soils are associated with higher break risk. The pipe-intrinsic variables (diameter, length, year installed, and age) behave the same for AC as they do for CI, with larger mains at reduced risk, longer mains at increased risk, and more recently installed mains at reduced risk. Unexpectedly, the Euclidean distance to a hydrant was significant for AC mains, with greater distances from hydrants associated with higher break risks. It also appears that break risk decreases for AC mains farther from pump stations. Both time-varying and average surrogate water demand was not significant for AC main breaks.

The ST model has the fewest significant covariates of the models presented here. The only seasonal variable included in the model is average ambient air temperature, with increasing temperatures associated with reduced break risk. The covariates for diameter and length behave the same for ST as discussed previously for the other materials. However, diameter appears to be more important than length for ST and CI, while the opposite is true for AC. The age of ST mains has a positive coefficient, which is intuitive given that pipes degrade as they age. The cohort of ST pipes is much younger than the population of CI and AC pipes, so perhaps the change in coefficient sign for ST is due to observing the break risk during the beginning of a pipe's life rather than the end. The model suggests that gravity-fed ST mains are associated with lower break risk compared to pump-fed ST mains. The simulations accurately reflect the randomness of ST breaks and little to no seasonality is observed apart from the slight dependence on temperature, as shown in Figure 6.

Contrary to what the conceptual model (see Figure 1) assumes, the coefficient for age as a covariate is negative for CI and AC mains. Common sense holds that as mains age they experience and accumulate stress and consequently their risk of breaking increases. The graphical summary of the assumed failure

rates for various models as shown in Scheidegger et al. (2015) demonstrates this assumption clearly, as most of the theoretical failure rates increase monotonically with age. Le Gat (2014) gives an empirical pipe failure rate curve for ST core concrete pipes that demonstrates the initial high-risk phase, subsequent low risk phase, and eventual increase in risk due to aging. However, the break risk curve shown by Le Gat (2014) is not monotonic after the initial high-risk phase, as the older pipes sometimes experience years of significantly reduced break risk despite getting older. It is certainly expected that over the long term as mains age their failure rate will increase, which would suggest that the estimated parameter for age as covariate should be positive; however, the model presented here is limited to an eight-year period, which may not be long enough to accurately quantify the long-term effects of aging. Furthermore, the data shown in Figure 5 demonstrate that the failure risk may not monotonically increase with age. CI mains clearly have a peak and subsequent decrease in empirical break risk. This could the result of the older mains being sturdier or in a less stressful environment than those that failed earlier. And while AC and ST pipes mainly have increasing break risk as they age, they may in the future exhibit similar behavior to CI pipes. Thus, a more nuanced model (e.g., including a Boolean variable that indicates whether the pipe is past the age of peak break risk) or a longer study period may be required to adequately capture the effects of aging on main break risk.

3. Pipe-to-Break Assignment Algorithm

Main break data were provided by EBMUD in the form of a geospatial dataset containing the location and date reported for each break in addition to the attributes of the pipe on which the break occurred, the failure mode, comments by the maintenance crew, and other identifying information regarding the pipeline on which the break occurred. Additionally, a geospatial pipeline database was provided that gives the location and attributes of each main, here defined as the unit of pipe assigned a unique identification number by EBMUD in the provided datasets, including material, diameter, length, installation year, life cycle status, life cycle status change date, pressure zone, and others. However, there is no direct link between the main break dataset and the provided pipeline dataset, meaning we cannot assign breaks to pipes with full certainty. This means that the two datasets, one for breaks and one for mains, must be combined to obtain the best available estimate of the break history for each pipe segment and to infer missing pipes.

To obtain the best available estimate of each pipe's break history, the algorithm below matches breaks to pipes and infers missing pipes if necessary based on the available evidence. EBMUD has confidence in the geospatial fidelity of the mains to within approximately $\pm$ 1 ft and the main breaks are associated with the nearest address. The laterals, which connect mains and service connections (with addresses), are used to adjust the locations of the main breaks to overlap the mains. The main breaks are then assigned to the most likely nearest pipe based on matching pipe characteristics. If no suitable pipe is found, mains are inferred based on the pipe information found in the main break records.

Because the break data contain pipe attributes, missing pipes can be inferred, but this means that detection of these missing pipes is limited to those that broke (this is probably not a significant source of error, because if it did not break, it likely would not have been replaced). Another example of missing pipes involves those that are actively carrying water but haven't been recorded in the GIS data yet due to either the inevitable lag between installing the pipe and recording the pipe in the GIS data or due to a backlog of pipes that need to be entered into the GIS dataset. It can take anywhere from months to years for a newly installed pipe to be placed into the GIS records.

Fortunately, the main breaks and mains can be related geospatially, though this presents a host of other challenges and uncertainties due to inaccuracies present in the GIS data, the most intractable example of

which is that there are mains that are not shown as existing according to the GIS data but in fact did exist and were simply replaced and deleted from the GIS records.

The main breaks are located by the nearest address, and since there is an automatically generated line representing the lateral pipe connecting the service address and the main, the main breaks can be shifted geospatially to overlap with the appropriate main, assuming it exists in the record. Multipoint features are created when one intersects the lines representing laterals and the lines representing mains to get the assumed adjusted locations of the main breaks. The breaks are on top of a service connection and the service connections are attached to the mains via lines representing laterals generated by EBMUD. Where those lines intersect the mains, one may generate points and assign the "adjusted" locations to the main breaks, which makes determining which pipe and which break go together significantly easier and less uncertain. Then one can reduce the search radius and be surer if you need to infer a pipe. Multiple points are produced when a lateral crosses two pipes, but that not an issue, since the algorithm only allows a single pipe per break.

Laterals can intersect several mains, however, this is not a problem since any main in the path of a lateral is a candidate for the break in question. The algorithm for assigning each break to a pipe ensures each break is only assigned to a pipe once.

To adjust the main break locations to be located on top of the mains instead of at the end of a lateral at a service address, the following workflow was used: find intersection between laterals and mains (this places points where laterals hit the mains), then do a near table to find which breaks are closest to which laterals, then assign the coordinates of the nearest lateral's intersection with its main to the breaks. This can result in a single main break being duplicated when the lateral intersects several pipes. This is taken care of in the algorithm for assigning each break to each pipe.

The method of joining the break data to the mains data involves creating a "near table" in ArcMap where each break is matched with the 10 nearest pipes within 100 feet. The attributes of both separate datasets are joined to this table and subsequently used to infer which break belongs to which pipe using the algorithm below.

To run any pipe-level point process-based algorithm, one needs the set of all pipes and the break history for each pipe. The algorithm used to produce this necessary combined dataset is as follows.

1. Algorithm for combining the mains dataset with the main break dataset into the best available estimate of the data we need.
    a. Find 10 nearest mains for each main break within 50 ft (using geospatial software to produce a "near table", for example, ArcMap)
    b. Join the break and pipe attributes to the near table. The break data contains information about the relevant pipeline, so we check if the pipeline attributes in the break data match the attributes of the nearest mains according to the pipeline dataset.
    c. Python script loops over each row and checks if:
        i. Materials match
        ii. Break date >= Install Year (from pipe database, here we are giving the pipe database priority over the break database in terms of the install date of the pipe)
        iii. Break date < abandon date, if abandoned or not in use
        iv. Install year of the pipe according to the break dataset is the same as install year of the pipe according to the pipeline dataset.
            1. This may be problematic if there are many cases where the two do not match. The analyst must use their bust judgement as to which source is

more reliable. Here we assumed the break data were less reliable than the pipe database. Common problems are shown below as examples.

a. The break data has some years with no '19' or '20', for example, installed in '47'.
    i. Assume these are old records and old pipes, so assume 20th century for all
b. Two pipes have a value that could be 21st century, meaning none are '10', for example. They are both cast iron, which wasn't installed in the 21st century, so they have to be '19' + 'xx'.
c. There are two '190's in the data. Assume this means 1990. There are very few pipes from 1890.

v. Check if all above are met
vi. Of the candidates (all conditions met), select the closest pipe to the break (lowest rank)
vii. There are others that are breaks on a pipe that has been removed but has not been placed into the GIS data. These breaks must be assigned to a new pipe, a new pipe that assumes the same location and length as the "near" pipe according to the table, but with the break's reported pipe install year and material type and other properties. This is the case when the break occurs before the install date of the nearest pipe when there are no other candidate pipes that satisfy earlier assignment.
    1. Check that break occurs before pipe's install date
    2. Of all candidate pipes, choose closest to break
    3. Generate new pipe using the pipe data found in the break data instead of the pipe data found in the pipe GIS data.
viii. Loop through the flagged data with inferred pipes marked and concatenated to the rest of the regular data, create new database from each row, obtaining pipe characteristics from the break values for inferred pipes and from the pipe database for non-inferred pipes.
ix. Assume that the old pipe was replaced when the new pipe was installed

A total of 3970 pipes were excluded due to lack of information (out of approximately 200,000 pipes).

4. Estimation Algorithm

SSPPs can be characterized by their conditional intensity function, which estimates the event occurrence rate at any point in space and time. SSPPs typically map to two-dimensional Euclidean space, although significant progress has been made recently in their application to linear networks (Baddeley et al., 2021). Because this model applies to a network of pipes whose composition is variable over time (pipes can be installed or abandoned/removed and occupy the same physical location at the same time), the spatial coordinates refer to each unique pipe (i), with spatial information associated with each pipe included as covariates. Each unique pipe (i) consists of a continuous segment of pipe as found in the geospatial pipe dataset provided by EBMUD. The model presented here is a self-exciting spatiotemporal point process with pipes $i \in P = \mathbb{Z}[0, n]$, events on pipes $j \subset P$ with event times $t_{j,k} \in \mathbb{R}[0, T + 1)$ (where T is the last time step in the study) with event number $k \in \mathbb{Z}[0, \infty]$, time steps $t \in \mathbb{Z}[0, T]$, and pipe history $\mathcal{H}_i$. The event times are not restricted to the piecewise linear domain of the set of time varying covariates. For example, one might observe the sequence of breaks shown in Table S1 with five total pipes ($P = \mathbb{Z}[0,4]$).

Table S1. Example of a series of pipe failures.

j	t_j	k
0	2.23	0
0	4.02	1
2	0.40	0
2	2.30	1
2	3.67	2
4	6.91	0
4	7.42	1
4	9.09	2
4	11.34	3

The conditional intensity for pipe i at time t given its history $\mathcal{H}_i$ takes the form given in Equation S1.

$$\lambda(i,t \mid \mathcal{H}_i) = \mu(i,t) + \begin{cases} \sum_{k=0}^{t_{i,k}<t} g\left(t - t_{i,k}\right), & \text{if } \exists\ t_{i,k} < t \\ 0, & \text{otherwise} \end{cases} \tag{S1}$$

Where $\mu(i,t)$ is the background intensity as shown in Equation S2

$$\mu(i,t) = \exp\left(\sum_{k=0}^{n_1} \beta_k^{\tau} X_{i,k}^{\tau}(t) + \sum_{k=0}^{n_2} \beta_k X_{i,k}\right) \tag{S2}$$

Where β^{τ} and β are vectors of n_1 and n_2 parameters, respectively, and X^{τ} and X are matrices of time-variant and time-invariant covariates, respectively. The triggering function $g(t - t_{j,k})$ accounts for self-excitation of breaks on the same pipe and is commonly specified in the exponential form found in Equation S3 (Laub et al., 2015).

$$\sum_{k=0}^{t_{j,k}<t} g\left(t - t_{j,k}\right) = \sum_{k=0}^{t_{j,k}<t} \alpha \mathrm{e}^{-\gamma\left(t - t_{j,k}\right)} \tag{S3}$$

Where α controls the increase in intensity due to an event and γ controls the decay. This model assumes a background intensity based on spatiotemporal covariates and a self-exciting triggering intensity that decays exponentially in time, a specification also assumed by Reinhart and Greenhouse (2018), although with the addition of a Gaussian spatial triggering term not considered here. Thus, the SSPP model fit here is given in full by Equation S4, with $\mathcal{H}_i$ implied hereafter.

$$\lambda(i,t) = \exp\left(\sum_{k=0}^{n_1} \beta_k^{\tau} X_{i,k}^{\tau}(t) + \sum_{k=0}^{n_2} \beta_k X_{i,k}\right) + \begin{cases} \sum_{k=0}^{t_{i,k}<t} \alpha \mathrm{e}^{-\gamma\left(t - t_{i,k}\right)}, & \text{if } \exists\ t_{i,k} < t \\ 0, & \text{otherwise} \end{cases} \tag{S4}$$

The parameters of self-exciting spatiotemporal point process models are estimated by maximizing the log-likelihood, which is adapted from Reinhart (2018) as shown in Equation S5. Maximization is performed using either Sequential Least Squares Programming (SLSQP) or the modified Powell algorithm as implemented in the Python package SciPy (Virtanen et al., 2020).

$$l(\widehat{\Theta}) = \sum_{t_j} \ln\left[\lambda(j, t_j)\right] - \int_0^T \int_P \lambda(i,t) didt \quad \text{(S5)}$$

Maximization of Equation S5 was performed using scipy.optimize.minimize(method='SLSQP'), or sequential least squares programming. An extremely useful check to make sure your maximization is proceeding as it should is the fact that the second half of Equation S5 should equal the number of breaks in the sample being analyzed, as it is an integral of the break risk over all the pipes and over all of their lifetimes. For example, the second half of Equation S5 should equal 2362 at the end of the maximization routine if the number of elements in j_array (as defined below) is 2362 (meaning you have 2362 total breaks under consideration).

The first half of Equation S5, called LL_I here, is the sum of the log of the conditional intensity evaluated at each event j.

So,

$$LL_I = \sum_{t_j} \ln\left[\lambda(j, t_j)\right] = \sum_{t_j} \ln\left[\exp\left(\sum_{k=0}^{n_1} \beta_k^{\tau} X_{j,k}^{\tau}(t_j) + \sum_{k=0}^{n_2} \beta_k X_{j,k}\right) + \sum_{k=0}^{t_{j,k}<t_j} \alpha e^{-\gamma(t_j - t_{j,k})}\right]$$

The algorithm used to calculate this value and the necessary inputs (Table S2) are found below.

Table S2. Inputs to the estimation algorithm.

Variable Name	Data Type	Shape	Description
j_array	Integer	1-D array	List of all breaks by pipe ID, index is implicitly connected to tj_array
tj_array	Float	1-D array	List of all break times, index matches j_array
alpha	Float	Scalar	Triggering break risk increase parameter
gamma	Float	Scalar	Triggering decay parameter
Bt	Float	1-D array	Time-varying covariate coefficients
B	Float	1-D array	Static covariate coefficients
Xt	Float	3-D array (time,var,pipeid), for example, (192,5,34000) means 192 months, 5 covariates, and 34000 pipes.	Time-varying covariates; everything is monthly, so with continuous break times this means all covariates are step-wise. When accessing this data with respect to a given break time, you must use something like np.floor(x) to get an integer for indexing to get the right month.
X	Float	2-D array (var,pipeid)	Static covariates

Algorithm for LL_I:

To evaluate the conditional intensity at each event j, take the natural log, and sum the results.

To loop over each break j and evaluate the conditional intensity, one needs to know how many breaks have occurred on this same pipe prior to the current event being analyzed. A Boolean filter is not an option in Cython, which is needed to ensure the analysis does take excessive amounts of time to run. Instead, one can loop over all of j again inside the outer loop over j and check if the pipe ID matches (j) and if it is a match one may check whether the time (tj) is before the event under investigation. If the

break is before the current event and is on the same pipe, one can evaluate the trigger (however, one should multiply by alpha at end instead of over and over in the loop) and add to TrigSum, which is reset to zero for every j in the outer loop over j. Thus, one may sum all of the trigger contributions and multiply by alpha. Then one may sum the background intensity into MuSum, then exponentiate MuSum (exp(MuSum)), then one may sum those both together, then take the natural log and add it to the running total for LL_I.

Preprocessing: sort j_array ascending and make sure tj_array is sorted according to j_array, this will allow for a shortcut later.

1. Start for loop for index of each element of j_array, index = u
    1.1. Set MuSum = 0
    1.2. Set TrigSum = 0
    1.3. Start for loop for index of each element of j_array, index = k
        1.3.1. Check if j_array[k]==j_array[u], meaning now the inner loop is on a break that is on the same pipe as is being examined in the outer loop
            1.3.1.1. Check if tj_array[k]<tj_array[u], meaning check if the found break on the same pipe is before the break being examined in the outer loop
                1.3.1.1.1. If this is the case, then we need to get the triggering contribution, so evaluate triggering function with the delta t and add to TrigSum
                1.3.1.1.2. TrigSum += exp(-gamma*(tj_array[u]-tj_array[k]))
                1.3.1.1.3. Don't multiply by alpha yet, can do that later to speed up the code
            1.3.1.2. Check if j_array[k]>j_array[u], if this is the case, one does not need to look at any more breaks in this inner loop since the lists were sorted ascending and if this is true then there won't be any more breaks on this pipe for the rest of j_array, so break the loop and go the next k
    1.4. Start for loop for index of each element of Bt
        1.4.1. Add to MuSum the multiplication of each Bt with the corresponding Xt value where t = tj_array[u]
    1.5. Start for loop for index of each element of B
        1.5.1. Add to MuSum the multiplication of each B with the corresponding X value where t = tj_array[u]
    1.6. Take natural log of (Trigsum*alpha + the exponential of MuSum)
    1.7. Add to LL_I

The second half of Equation S5, called LL_{II} here, is the numerically challenging part of this calculation. It is the integral of the conditional intensity over all time and over all pipes. Not all pipes are in existence over all time, so the bounds of integration for time change based on which pipe is being evaluated. A pipe can only contribute to the integral if it exists at time t. For this analysis, dt = 1 month, and each pipe is given the same weight, so one should evaluate and sum, leaving the multiplication by 1 as implicit for the integrals.

So,

$$LL_{II} = \int_0^T \int_P \lambda(i,t) didt = \int_0^T \int_P \left[\exp\left(\sum_{k=0}^{n_1} \beta_k^\tau X_{i,k}^\tau(t) + \sum_{k=0}^{n_2} \beta_k X_{i,k} \right) + \begin{cases} \sum_{k=0}^{t_{i,k}<t} \alpha e^{-\gamma(t-t_{i,k})}, & \text{if } \exists\ t_{i,k} < t \\ 0, & \text{otherwise} \end{cases} \right] didt$$

$$= \int_0^T \int_P \left[\exp\left(\sum_{k=0}^{n_1} \beta_k^\tau X_{i,k}^\tau(t) + \sum_{k=0}^{n_2} \beta_k X_{i,k} \right) \right] didt + \int_P \int_{t_1}^{t_2} \left[\begin{cases} \sum_{k=0}^{t_{i,k}<t} \alpha e^{-\gamma(t-t_{i,k})}, & \text{if } \exists\ t_{i,k} < t \\ 0, & \text{otherwise} \end{cases} \right] didt$$

$$= \sum_i \sum_{t=t_{i,start}}^{t_{i,end}} \left[\exp\left(\sum_{k=0}^{n_1} \beta_k^\tau X_{i,k}^\tau(t) + \sum_{k=0}^{n_2} \beta_k X_{i,k} \right) \right] + \sum_i \sum_{t_{i,k}} \frac{\alpha}{\gamma} \left(e^{\gamma(t_{i,k}-t_{i,k})} - e^{\gamma(t_{i,k}-t_{i,end})} \right)$$

$$= \sum_i \sum_{t=t_{i,start}}^{t_{i,end}} \left[\exp\left(\sum_{k=0}^{n_1} \beta_k^\tau X_{i,k}^\tau(t) + \sum_{k=0}^{n_2} \beta_k X_{i,k} \right) \right] + \sum_i \sum_{t_{i,k}} \frac{\alpha}{\gamma} \left(1 - e^{\gamma(t_{i,k}-t_{i,end})} \right)$$

The additional inputs for this algorithm are shown in Table S3.

Table S3. Additional inputs.

Variable Name	Data Type	Shape	Description
d_i_array	Integer	1-D array	List of all pipes considered, not just those with breaks
d_start_array	Float	1-D array	List of all start times for the pipes in d_i_array
d_end_array	Float	1-D array	List of all end times for the pipes in d_i_array

Because the trigger and the background are additive, they are separable. To speed up the code by taking advantage of some tricks, one may evaluate them separately. The trigger is always zero for the majority of the pipes, so one should not waste time looping over them. One should check for pipes with breaks to count. Additionally, we recommend not discretizing the integral over all of the time periods, rather, integrate the function and use the result.

1. First evaluate the background part of the integral
    1.1. Loop over index of each pipe in d_i_array, idx = u
        1.1.1. Loop over index of each time step in range from t start to t end for that pipe
            1.1.1.1. Define MuSum = 0
            1.1.1.2. Add up the Bt*Xt to MuSum
            1.1.1.3. Add up the B*X to MuSum
            1.1.1.4. Add the exponential of MuSum to LL_{II}
2. Now evaluate the trigger
    2.1. Loop over index of each pipe in d_i_array, idx = u
        2.1.1. Loop over index of each break in j_array, idx = k
            2.1.1.1. Check if pipe given by d_i_array[u] matches pipe given by j_array[k]
                2.1.1.1.1. If match, get trigger integral value from tj_array[k] to d_End_array[u]
                    2.1.1.1.1.1. $\frac{\alpha}{\gamma}\left(1 - e^{\gamma(t_{i,k}-t_{i,end})}\right)$
                2.1.1.1.2. LL_{II}+= value

5. Estimation Algorithm – Scheidegger et al.'s (2013) Model

The conditional intensity, or break risk for pipe i at time t, where n is the previous number of breaks at time t, is given by Equation S6.

$$\lambda^*(i,t,n) = \begin{cases} \theta_1\theta_2(\theta_2 t)^{\theta_1-1} & n(i,t)=0 \\ \theta_3, & n(i,t)>0 \end{cases} \quad \text{(S6)}$$

Note the time to the first failure is assumed to follow the Weibull distribution. To incorporate covariates into the equation, the following method is used as outlined in Scheidegger et al. (2015).

$$\lambda(i,t,n) = \left(\begin{cases} \theta_1\theta_2(\theta_2 t)^{\theta_1-1} & n(i,t)=0 \\ \theta_3, & n(i,t)>0 \end{cases} \right) \times \exp\left(\sum_{k=0}^{n_1} \beta_k^\tau X_{i,k}^\tau(t) + \sum_{k=0}^{n_2} \beta_k X_{i,k} \right)$$

The parameters (thetas and betas) may be estimated by maximizing the log-likelihood, adapted from Reinhart (2018) as noted previously in Equation S5.

The first half of Equation S5, called LL_I here, is the sum of the log of the conditional intensity evaluated at each event j.

So,

$$LL_I = \sum_{t_j} \ln\left[\lambda(j,t_j)\right] = \sum_{t_j} \ln\left[\left(\begin{cases} \theta_1\theta_2(\theta_2 t_j)^{\theta_1-1} & n(j,t)=0 \\ \theta_3, & n(j,t)>0 \end{cases} \right) \times \exp\left(\sum_{k=0}^{n_1} \beta_k^\tau X_{j,k}^\tau(t_j) + \sum_{k=0}^{n_2} \beta_k X_{j,k} \right) \right]$$

Algorithm for LL_I:

1. Start for loop for index of each element of j_array, index = u
   1.1. Set CoefSum= 1
   1.2. Start for loop for index of each element of Bt
      1.2.1. Add to CoefSum the multiplication of each Bt with the corresponding Xt value where t = tj_array[u]
   1.3. Start for loop for index of each element of B
      1.3.1. Add to CoefSum the multiplication of each B with the corresponding X value where t = tj_array[u]
      1.3.2. Take natural log of (theta1 *theta2*(theta2*tj)**(theta1-1))*CoefSum
   1.4. Add to LL_I

The second half of Equation S5, called LL_{II} here, is the numerically challenging part of this calculation. It is the integral of the conditional intensity over all time and over all pipes. Not all pipes are in existence over all time, so the bounds of integration for time change based on which pipe is being evaluated. A pipe can only contribute to the integral if it exists at time t. For this analysis, dt = 1 month, and each pipe is given the same weight, so one should just evaluate and sum, leaving the multiplication by 1 as implicit for

the integrals. All terms are evaluated numerically here, whereas before the triggering term was integrated analytically to yield a more efficient algorithm.

So,

$$LL_{II} = \int_0^T \int_P \lambda(i,t) didt = \int_0^T \int_P \left(\begin{cases} \theta_1\theta_2(\theta_2 t)^{\theta_1 - 1} & n(t) = 0 \\ \theta_3, & n(t) > 0 \end{cases} \right) \times \exp\left(\sum_{k=0}^{n_1} \beta_k^\tau X_{i,k}^\tau(t) + \sum_{k=0}^{n_2} \beta_k X_{i,k} \right) didt$$

$$= \sum_i \sum_{t=t_{i,start}}^{t_{i,end}} \left[\left(\begin{cases} \theta_1\theta_2(\theta_2 t)^{\theta_1 - 1} & n(t) = 0 \\ \theta_3, & n(t) > 0 \end{cases} \right) \times \exp\left(\sum_{k=0}^{n_1} \beta_k^\tau X_{i,k}^\tau(t) + \sum_{k=0}^{n_2} \beta_k X_{i,k} \right) \right]$$

The additional inputs for this algorithm are as follows.

1. Loop over index of each pipe in d_i_array, idx = u
   1.1. Loop over index of each time step in range from t start to t end for that pipe
      1.1.1. Define CoefSum = 0
      1.1.2. Add up the Bt*Xt to CoefSum
      1.1.3. Add up the B*X to CoefSum

6. Estimation Algorithm for Lin and Yuan's (2019) Model

The version of Lin and Yuan (2019) used here is given by the conditional intensity function for pipe i at time t given in Equation S7.

$$\lambda_L(i,t,w) = \alpha \left(\prod_{k=0}^{n_1} \left(X_{i,k}^\tau(t)\right)^{\beta_k^\tau} \prod_{k=0}^{n_2} \left(X_{i,k}\right)^{\beta_k} \right) \times t^\eta \times \begin{cases} \left(t - max(t_{i,0:k} < t)\right)^\psi, & \text{if } \exists\ t_{i,k} < t \\ 1, & \text{otherwise} \end{cases} \quad \text{(S7)}$$

The parameters (thetas and betas) may be estimated by maximizing the log-likelihood, adapted from Reinhart (2018) as noted previously in Equation S5. The first half of Equation S5, called LL_I here, is the sum of the log of the conditional intensity evaluated at each event j.

$$LL_I = \sum_{t_j} \ln\left[\lambda(j,t_j)\right]$$

$$= \sum_{t_j} \ln \left[\alpha \left(\prod_{k=0}^{n_1} \left(X_{i,k}^\tau(t)\right)^{\beta_k^\tau} \prod_{k=0}^{n_2} \left(X_{i,k}\right)^{\beta_k} \right) \times t_j^\eta \times \begin{cases} \left(t_j - max(t_{j,0:k} < t_j)\right)^\psi, & \text{if } \exists\ t_{j,k} < t_j \\ 1, & \text{otherwise} \end{cases} \right]$$

To calculate, use the following algorithm.

1. Start for loop for index of each element of j_array, index = u
   1.1. Set MuSum = 1
   1.2. Set First = 1

1.3. Set LastBreak = 99999
1.4. Start for loop for index of each element of j_array, index = k
    1.4.1. Check if j_array[k]==j_array[u], meaning now the inner loop is on a break that is on the same pipe as is being examined in the outer loop
        1.4.1.1. Check if tj_array[k]<tj_array[u], meaning check if the found break on the same pipe is before the break being examined in the outer loop
            1.4.1.1.1. Check if this is the first prior break discovered, if it is, set LastBreak to tj_array[k] and set First = 0; if this is not the first prior break discovered, check if tj_array[k] > LastBreak, if it is, set LastBreak = tj_array[k]
        1.4.1.2. Check if j_array[k]>j_array[u], if this is the case, one does not need to look at any more breaks in this inner loop since the lists were sorted ascending and if this is true then there won't be any more breaks on this pipe for the rest of j_array, so break the loop and go the next k
1.5. Start for loop for index of each element of Bt
    1.5.1. Multiply MuSum by Xt^Bt where t = tj_array[u]
1.6. Start for loop for index of each element of B
    1.6.1. Multiply MuSum by X^B where t = tj_array[u]
    1.6.2.
1.7. Check if LastBreak = 99999
    1.7.1. If it does, LL_I += (alpha*MuSum*age**eta)
    1.7.2. If it does not, LL_I += alpha*MuSum*age**eta*(tj_array[u]-LastBreak)**psi

The second half of Equation S5, called LL_{II} here, is the numerically challenging part of this calculation. It is the integral of the conditional intensity over all time and over all pipes. Not all pipes are in existence over all time, so the bounds of integration for time change based on

which pipe is being evaluated. A pipe can only contribute to the integral if it exists at time t. Additionally, dt = 1 month, and each pipe is given the same weight, so should evaluate and sum, leaving the multiplication by 1 as implicit for the integrals.

So,

$$LL_{II} = \int_0^T \int_P \lambda(i,t)didt$$

$$= \int_0^T \int_P \left[\alpha \left(\prod_{k=0}^{n_1} \left(X_{i,k}^{\tau}(t)\right)^{\beta_k^{\tau}} \prod_{k=0}^{n_2} \left(X_{i,k}\right)^{\beta_k} \right) \times t^{\eta} \right.$$

$$\left. \times \begin{cases} \left(t - max(t_{i,0:k} < t)\right)^{\psi}, & \text{if } \exists\ t_{i,k} < t \\ 1, & \text{otherwise} \end{cases} \right] didt$$

$$= \sum_i \sum_{t=t_{i,start}}^{t_{i,end}} \left[\alpha \left(\prod_{k=0}^{n_1} \left(X_{i,k}^{\tau}(t)\right)^{\beta_k^{\tau}} \prod_{k=0}^{n_2} \left(X_{i,k}\right)^{\beta_k} \right) \times t^{\eta} \times \begin{cases} \left(t - max(t_{i,0:k} < t)\right)^{\psi}, & \text{if } \exists\ t_{i,k} < t \\ 1, & \text{otherwise} \end{cases} \right]$$

To calculate the value, use the following algorithm.

1. Loop over index of each pipe in d_i_array, idx = u
2. Set First = 1
3. Set LastBreak = 99999
    3.1. Loop over index of each time step in range from t start to t end for that pipe
        3.1.1.Define MuSum = 1
        3.1.2.Multiply MuSum by Xt^Bt for each covariate
        3.1.3.Multiply MuSum by X^B for each covariate
        3.1.4. Loop over index of each break in j_array, idx = k
            3.1.4.1. Check if pipe given by d_i_array[u] matches pipe given by j_array[k]
                3.1.4.1.1. If match, check if tj_array[k] is less than t
                    3.1.4.1.1.1. If first one that is, make LastBreak = tj_array[k]
                    3.1.4.1.1.2. If not, check if tj_array[k] > LastBreak
                        3.1.4.1.1.2.1. If yes, make LastBreak = tj_array[k]
                3.1.4.1.2. If tj_array[k] > t, break loop. Because tj_array is sorted ascending there will be no more possible matches.
        3.1.5.Check if LastBreak = 99999
            3.1.5.1. If it does, LL_II += (alpha*MuSum*age**eta)
            3.1.5.2. If it does not, LL_II += alpha*MuSum*age**eta*(t-LastBreak)**psi

7. Simulation Algorithm for Stepwise Forecasting

We want to make forecasts based on the best available true history. When you simulate for a long period, the error propagates. Updating the simulated history with the true history can significantly improve forecasts. For the training period we use monthly to biannual forecasting intervals, for the testing period, we use a single simulation for the entire period but use the true history for the entire training period.

This simulation method generates the background events using Lewis' thinning algorithm (Lewis and Shedler, 1979; Zhuang and Touati, 2019) and generates the children events using Algorithm 5 in Reinhart (2018), which is also found in Zhuang et al. (2004).

If you would like to use synthetic data to check your calculations, we recommend generating the time varying covariates using sinusoids to mimic seasonality and generating the time-invariant covariates using Gaussian distributions.

An important quantity to be used in the simulation, m, is calculated as follows.

1. $m = \int_X \int_0^T g(s,t)dtds$ according to Equation 3 in Reinhart (2018) and represents the mean number of offspring (children).
2. In this case, the locations of the children are not drawn from the trigger, only the break times for each pipe we are focusing on. In this case of the SSPP model, $m = \int_0^T g(t)dt$, which gives the average number of children per break per pipe.
3. Given that $g(t) = \alpha e^{-\gamma t}$, then $m = \int_0^T g(t)dt = \frac{\alpha e^{-\gamma T}}{-\gamma} - \frac{\alpha}{-\gamma} = \frac{\alpha}{\gamma} - \frac{\alpha e^{-\gamma T}}{\gamma} = \frac{\alpha}{\gamma}(1 - e^{-\gamma T})$

Generate events for each pipe based on the background intensity (μ) using Lewis' thinning algorithm.

1. Select pipe i
2. Create empty list for accepted events, "events_i"

3. Get maximum conditional intensity value for pipe i
    a. The majorizing function is selected so that it is always greater than or equal to the actual conditional intensity (take the maximum of the background function over all time steps).
        i. $\lambda_u = \max\left(\exp\left(\sum_{k_1=0}^{n_1} \beta_{k_1} X_{k_1}(t) + \sum_{k_2=0}^{n_2} \beta_{k_2} X_{k_2}\right)\right) = K$
        ii. The betas are constant between pipes, but the covariates are pipe-specific and change according to the pipe.
4. Generate the number of events as a Poisson random variable with mean λ_u multiplied by the time step for forecasting (variable = events)
5. For each event time, generate a uniformly distributed event time between the start of the forecast period and the end of the forecast period (np.random.uniform(low=t_{min},high = t_{max}, size = len(events))
6. For each event,
    a. Generate the background conditional intensity at time t (np.floor() applied to the event time) (background).
    b. Generate a value that is uniformly distributed between 0 and 1 (u).
    c. Accept event if u <= background/λ_u

Now that the background (parents, or generation 0) event have been generated according to the background conditional intensity function (nonhomogeneous Poisson, or NHPP, in the case of the SSPP model), the children are generated according to the triggering function g. This algorithm is adapted from Algorithm 5 in Reinhart (2018).

1. The events generated previously are an NHPP according to the background intensity, they are considered the parent generation (G), or generation 0, or $l = 0$
2. For each event j in $G(l)$ simulate the number of offspring as N~Poisson(m), where m is as defined previously.
3. For each child, get the event time by making a random draw from the probability distribution given by the triggering function.
    a. Normalized, the triggering function induces a probability distribution for the times of the offspring events.
        i. Normalized means dividing by m ($\frac{g(t)}{m}$)
        ii. Need to draw from $g(t)/m$'s cdf like we did with F earlier, with uniform samples
        iii. $cdf = \int_0^y \frac{g(t)}{m} dt = \int_0^y \frac{\alpha e^{-\gamma t}}{m} dt = \frac{-\alpha e^{-\gamma y}}{\gamma m} + \frac{\alpha}{\gamma m} = \frac{\alpha}{\gamma m}(1 - e^{-\gamma y})$
        iv. $F(y) = \frac{\alpha}{\gamma m}(1 - e^{-\gamma y}) \rightarrow y = -\frac{1}{\gamma}\ln\left(1 - \frac{\gamma m}{\alpha} F\right)$ where F ~ $U(0,1)$ and y is the event time of the child relative to the parent
    b. These offspring become the set $O_j(l)$, or the next generation of parents (so long as they don't exceed the study's end time).
4. Let $G(l+1) = O_j(l)$
5. Repeat until there are no more children.

The two algorithms are repeated for each pipe. The parents (background) are generated, then all of the children, then the next pipe's break behavior is simulated. All of the simulated data is aggregated together into a Python dictionary.

8. Break Risk as a Function of Pipe Age

The break risk as a function of pipe age (Figure 5) is computed as follows. The same methods were used to compute the break risk as a function of years since the previous break as shown in Figure 4. Consider the data in Table S4.

Table S4. Hypothetical pipe failure data.

	Pipe Number					
	1			2		
Year of Study	Length (m)	Age (yr)	Break (no=0 yes=1)	Length (m)	Age (yr)	Break (no=0 yes=1)
0	2	11	0	4	14	0
1	2	12	0	4	15	0
2	2	13	1	4	16	0
3	2	14	0	4	17	0
4	2	15	0	4	18	0
5	2	16	0	4	19	1
6	2	17	0	4	20	0

This data can be combined as shown in Table S5.

Table S5. Hypothetical pipe failure data.

Pipe No.	Age (yr)	Length (m)	Break
1	11	2	0
1	12	2	0
1	13	2	1
1	14	2	0
1	15	2	0
1	16	2	0
1	17	2	0
2	14	4	0
2	15	4	0
2	16	4	0
2	17	4	0
2	18	4	0
2	19	4	1
2	20	4	0

Grouping by age and summing the lengths and breaks yields the dataset shown in Table S6.

Table S6. Hypothetical failure data.

Age (yr)	Length (m)	Breaks
11	2	0
12	2	0
13	2	1
14	6	0
15	6	0
16	6	0
17	6	0
18	4	0
19	4	1
20	4	0

Dividing the number of breaks by the length of pipe at that age yields the break rate. Plotting the resulting value as a function of age yields the pipe break risk as a function of age.

9. Figures

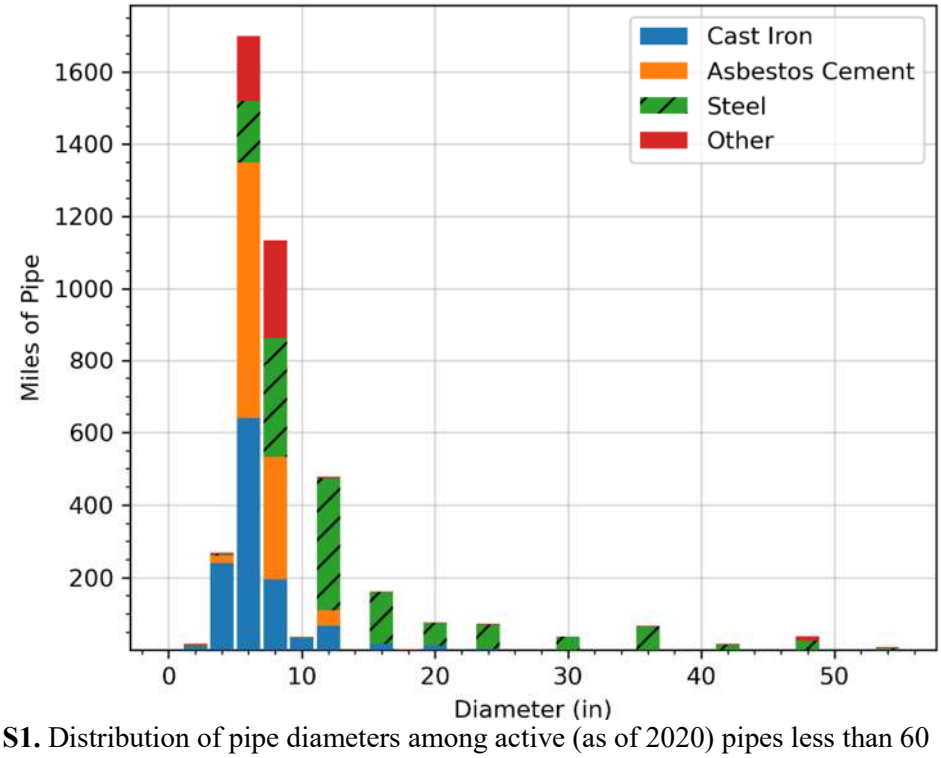

Fig. S1. Distribution of pipe diameters among active (as of 2020) pipes less than 60 inches in diameter.

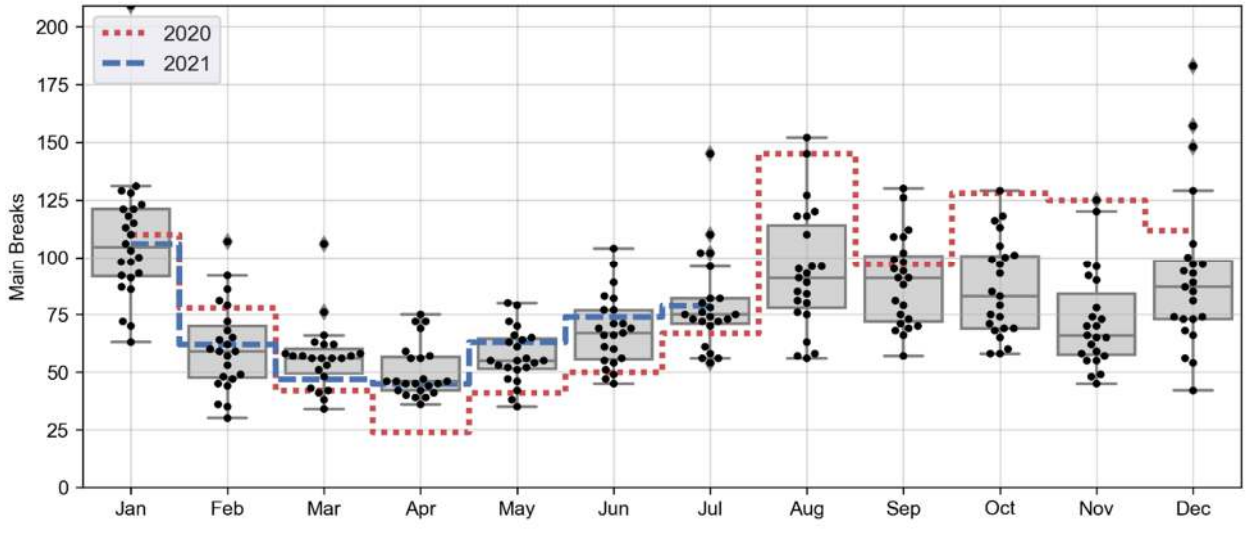

Fig. S2. Box and swarm plot of aggregated monthly main breaks from 1997-01-01 through 2019-12-31 (center line = median, edges of box = interquartile range, whiskers = rest of distribution, fliers = estimated to be outliers) with main breaks occurring in 2020 and 2021 shown as dashed lines.

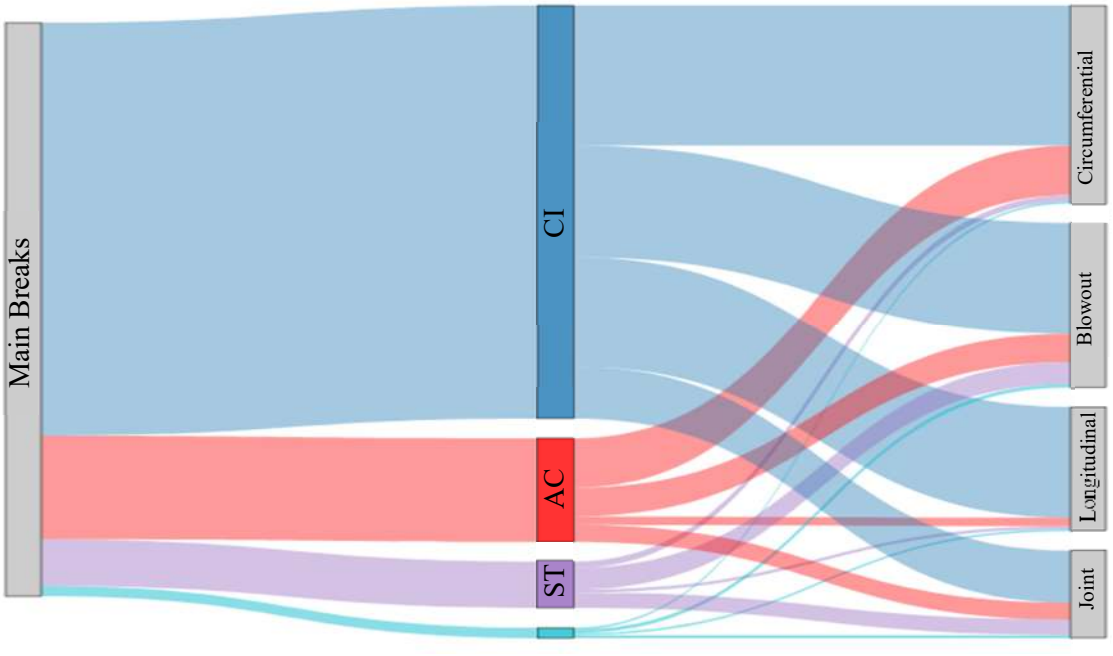

Fig. S3. Distribution of main breaks from 1997-01-01 through 2020-08-31 between material type and failure mode; the bottom (cyan) stream represents all other material types.

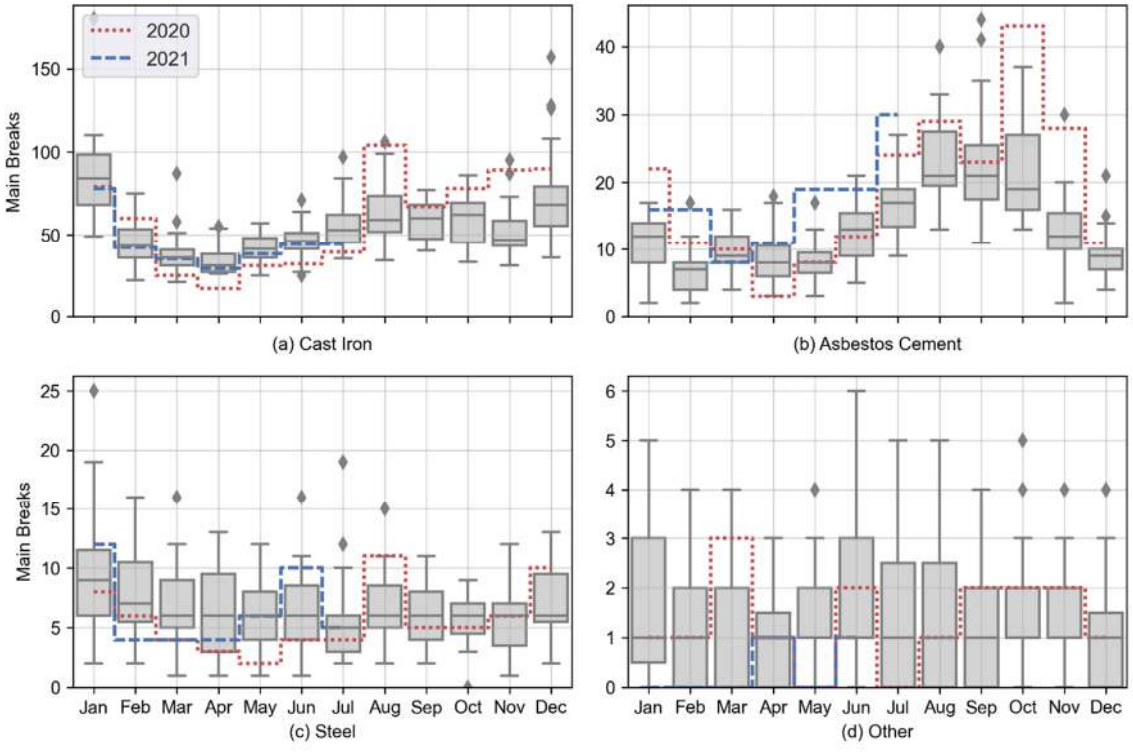

Fig. S4. Box plot of monthly main breaks from 1997-01-01 through 2019-12-31 with main breaks occurring in 2020 and 2021 shown as dashed lines. Note that the scales of the vertical axes are not identical.

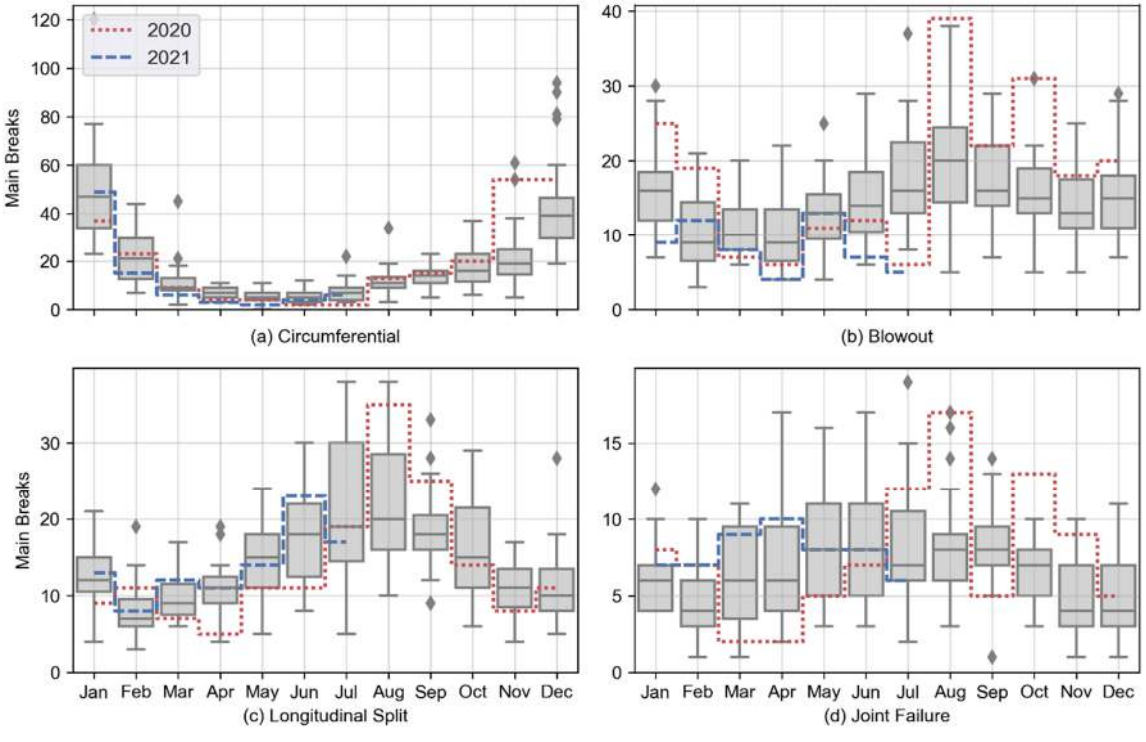

Fig. S5. Box plot of cast iron (CI) monthly total main breaks for 1997-01-01 through 2021-07-31 by failure mode with main breaks occurring in 2020 and 2021 shown as dashed lines. Note that the scales of the vertical axes are not identical.

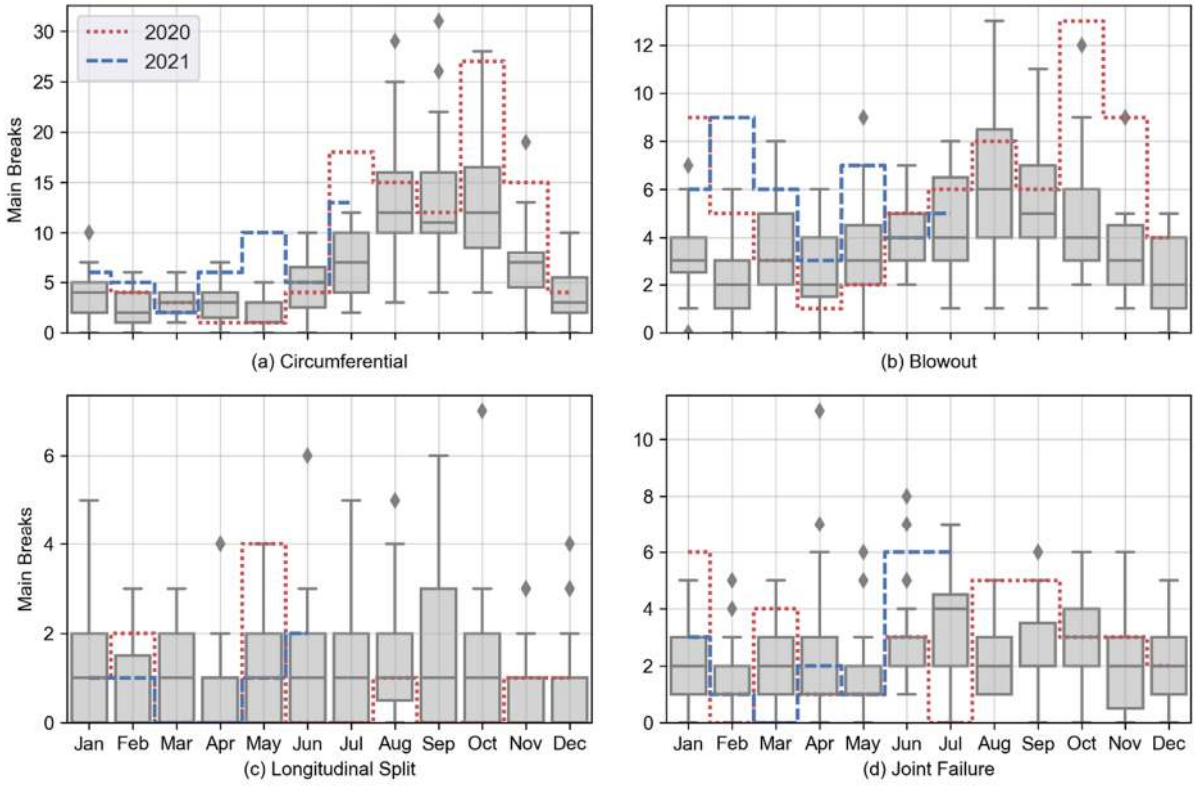

Fig. S6. Box plot of asbestos cement (AC) monthly total main breaks for 1997-01-01 through 2021-07-31 by failure mode with main breaks occurring in 2020 and 2021 shown as dashed lines.

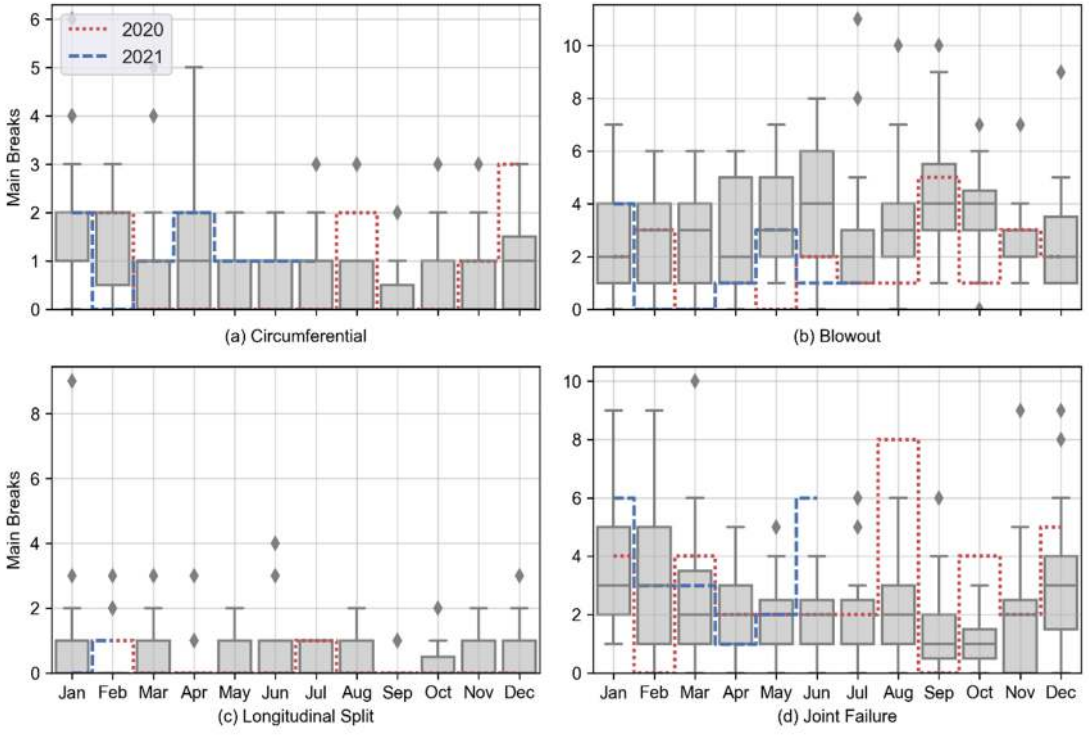

Fig. S7. Box plot of steel (ST) monthly total main breaks for 1997-01-01 through 2021-07-31 by failure mode with main breaks occurring in 2020 and 2021 shown as dashed lines. Note that the scales of the vertical axes are not identical.

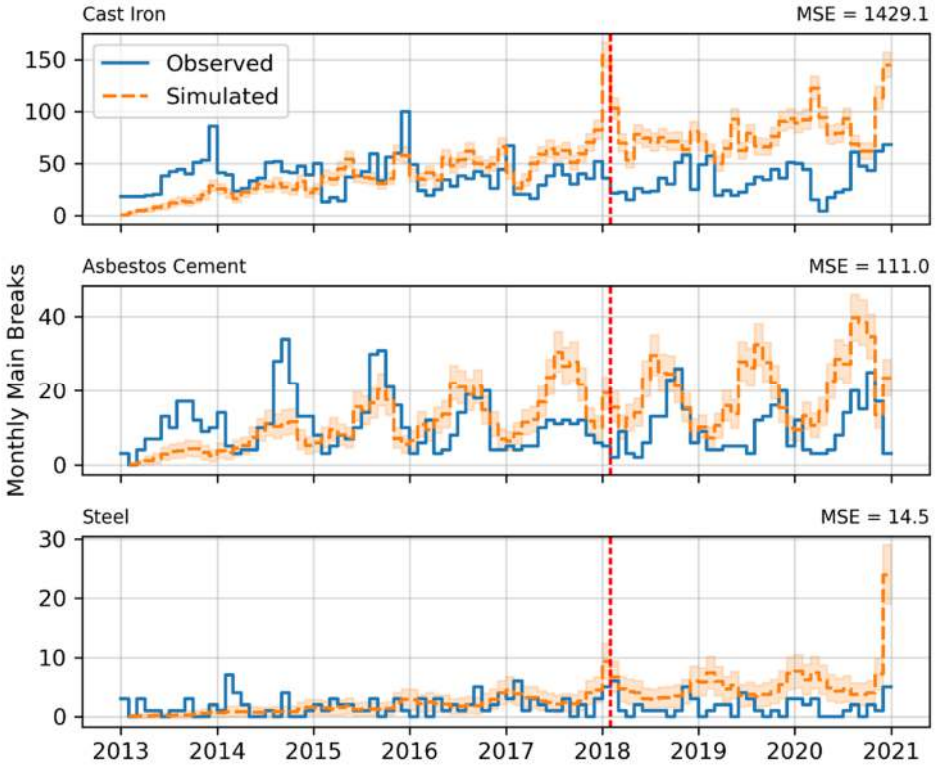

Fig. S8. Observed monthly main breaks vs 5,000 simulations using Scheidegger et al.'s (2013) model from 2013-01-01 through 2020-12-31 with the mean monthly simulated main breaks shown by the orange dashed line and the associated standard deviation given by the shaded region; the model was iteratively trained in monthly intervals to the left of the dashed red line and used to forecast the entire period to the right of the dashed red line and the mean square error (MSE) for the entire simulation is given for each material.

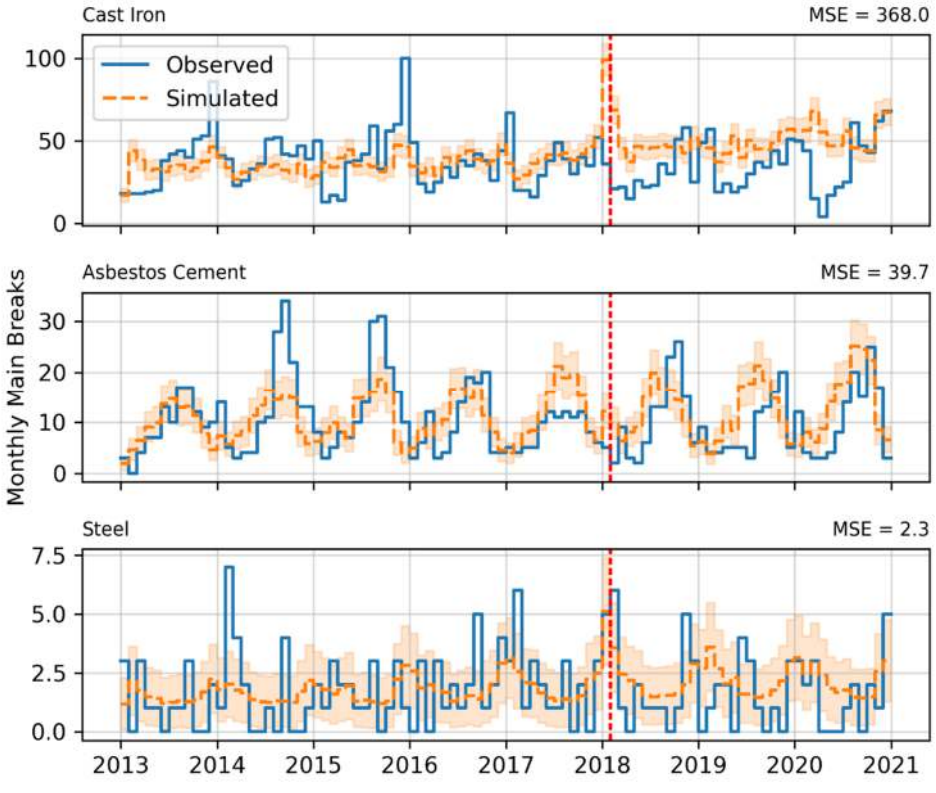

Fig. S9. Observed monthly main breaks vs 5,000 simulations using Lin and Yuan's (2010) model from 2013-01-01 through 2020-12-31 with the mean monthly simulated main breaks shown by the orange dashed line and the associated standard deviation given by the shaded region; the model was iteratively trained in monthly intervals to the left of the dashed red line and used to forecast the entire period to the right of the dashed red line and the mean square error (MSE) for the entire simulation is given for each material.

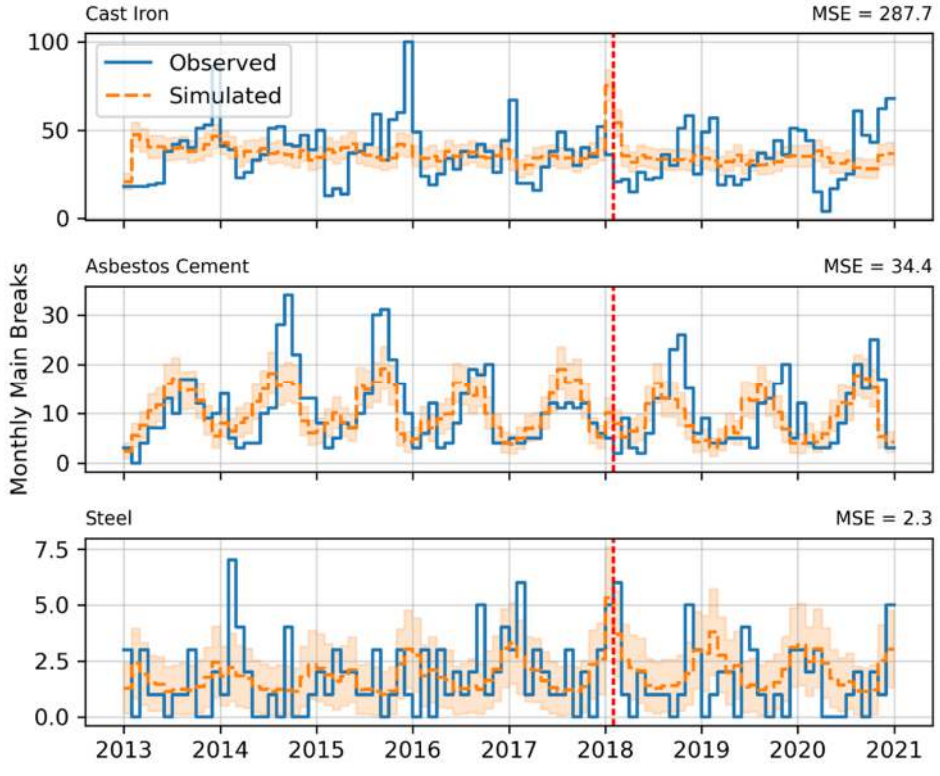

Fig. S10. Observed monthly main breaks vs 5,000 simulations using Kleiner and Rajani's (2010) model from 2013-01-01 through 2020-12-31 with the mean monthly simulated main breaks shown by the orange dashed line and the associated standard deviation given by the shaded region; the model was iteratively trained in monthly intervals to the left of the dashed red line and used to forecast the entire period to the right of the dashed red line and the mean square error (MSE) for the entire simulation is given for each material.

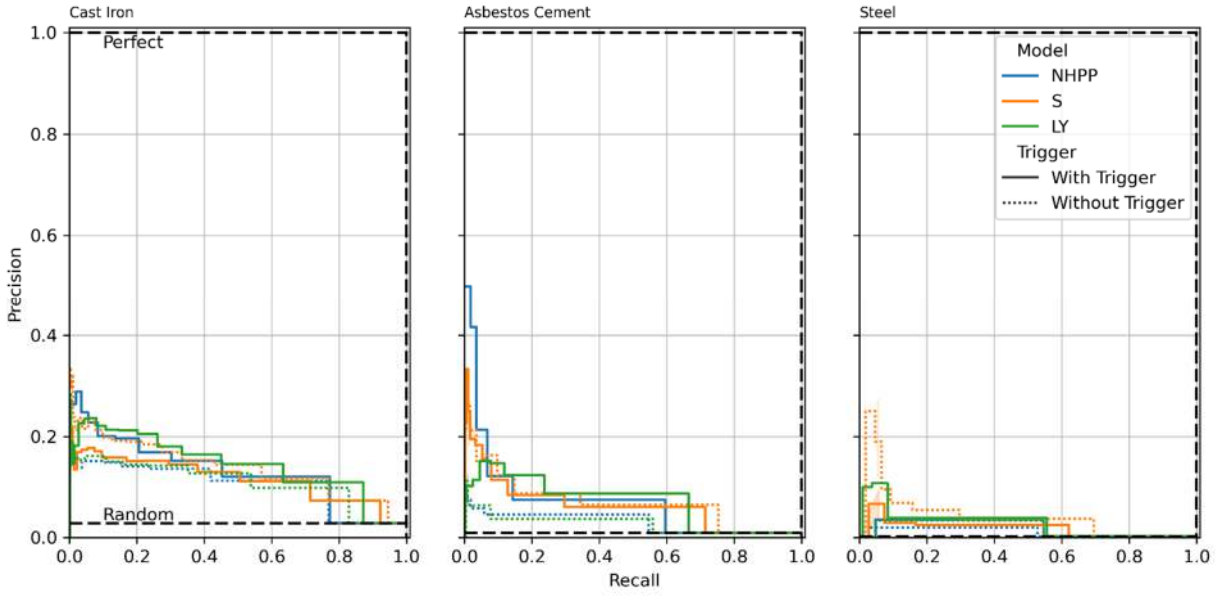

Fig. S11. Precision-recall curves for all models and materials. NHPP refers to the SSPP model and Kleiner and Rajani's (2010) model, because they both rely on a non-homogeneous Poison process (NHPP), S refers to Scheidegger et al.'s (2013) model, and LY refers to Lin and Yuan's (2019) model.

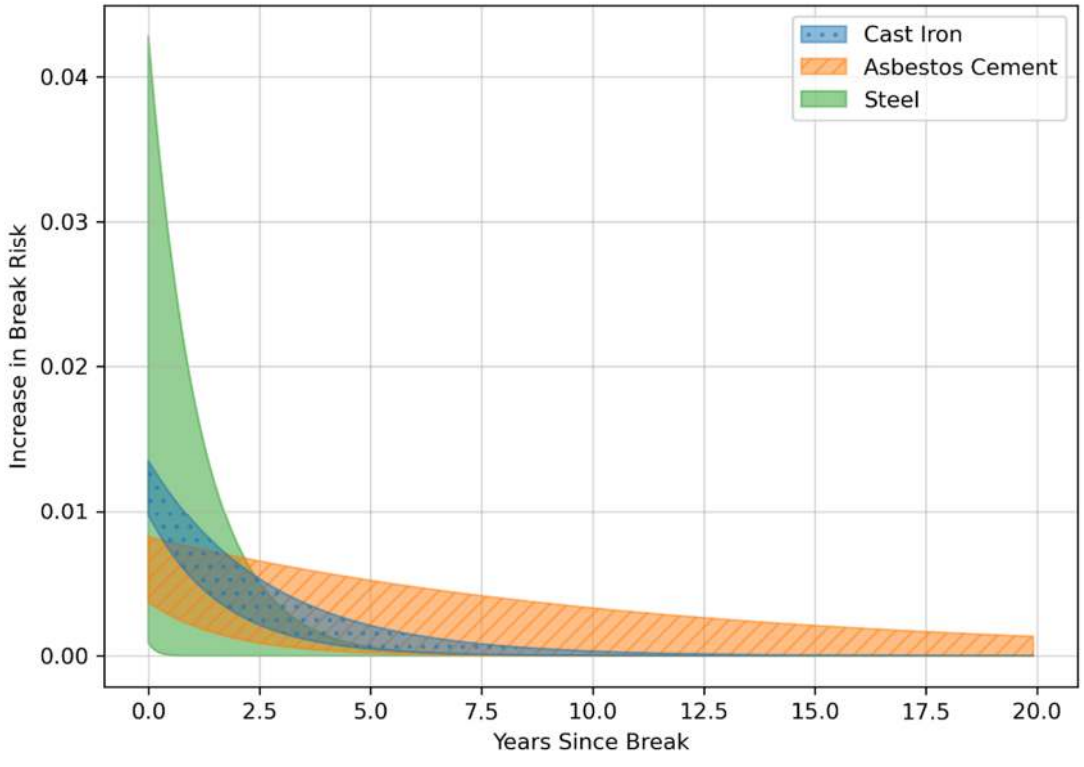

Fig. S12. Comparing triggering functions for each material type (95% confidence interval on the triggering parameters).

Table S7
Table of parameters for the fitted SSPP models.

Parameter	CI Estimate	CI 95% CI	AC Estimate	AC 95% CI	Steel Estimate	Steel 95% CI
α (immediate risk increase)	0.0116	0.0135, 0.0097	0.0060	0.0082, 0.0037	0.0219	0.0428, 0.0009
γ (risk decay constant)	0.0408	0.0506, 0.0311	0.0277	0.0477, 0.0076	0.2752	0.4791, 0.0712
Precipitation[a]	-0.8281	-0.4267, -1.230	-	-	-	-
Avg. Air Temperature	-0.9517	-0.5596, -1.344	2.702	3.169, 2.234	-1.957	-0.7284, -3.185
Water Demand	2.011	3.729, 0.2930	-	-	-	-
Age	-1.400	-0.7852, -2.015	-3.960	-2.921, -4.998	6.101	7.437, 4.766
Average Water Demand	-2.055	-0.2865, -3.823	-	-	-	-
Year Installed	-0.7805	-0.2079, -1.353	-4.350	-3.487, -5.214	-	-
Diameter	-5.094	-4.283, -5.884	-2.739	-1.609, -3.869	-17.63	-15.01, -20.24
Length	3.909	4.102, 3.717	3.858	4.258, 3.458	3.800	4.583, 3.017
Hydrant[c]	-	-	2.040	3.280, 0.7992	-	-
Pump Station[c]	-1.103	-0.7041, -1.501	-2.061	-1.312, -2.810	-	-
Gravity Fed[b]	-0.2234	-0.1061, -0.4307	-	-	-1.079	-0.4597, -1.699
Average Traffic[d]	-	-	-	-	-	-
Soil Clay Content	0.6192	0.8462, 0.3921	0.7254	1.167, 0.2840	-	-
Soil Electrical Conductivity	-	-	-	-	-	-
Soil pH	-	-	-1.006	-0.4013, -1.611	-	-

Notes: Covariates with "-" were excluded from the model due to insignificance or another similar variable being significant. Confidence intervals were used instead of P-values because they present the same information in a format that is easier to interpret.
[a]Two month rolling mean.
[b]Binary variable; alternative is pipe-fed.
[c]Euclidean distance.
[d]Traffic volume per square foot of grid area.

Table S8
Factors that influence pipe break rates.

Intrinsic	Extrinsic
Diameter	Water quality (pH, alkalinity, hardness, chlorine residual, etc.)
Material	Water pressure (operating and surge)
Cover depth	Seismic activity
Length	Differential settlement (attached structures, different soil types, etc.)
Age	Weather (precipitation, air temperature, drought conditions)
Wall thickness	Installation practices
Lining/coating	Cathodic protection
Young's Modulus	Water demand
Coef. of thermal expansion	Surface live loads (traffic, construction, etc.)
Vintage	Excavation accidents
Joint type	Maintenance and repair activities
Manufacturing practices	Corrosion
Manufacturing defects	Stray electrical currents
	Thrust restraint
	Nearby leaks
	Soil conditions (pH, electrical conductivity, redox potential, soil moisture, clay content, organic matter content, backfill and bedding material, chloride content, sulfate/sulfide content, dissolved oxygen, etc.)

Note: For more a more extensive review see Barton et al. (2019), Rajani and Kleiner (2001), and Folkman (2018).

References

Baddeley, A., Nair, G., Rakshit, S., McSwiggan, G., Davies, T.M., 2021. Analysing point patterns on networks — A review. Spat. Stat. 42, 100435. https://doi.org/10.1016/j.spasta.2020.100435.

Barton, N.A., Farewell, T.S., Hallett, S.H., Acland, T.F., 2019. Improving pipe failure predictions: Factors affecting pipe failure in drinking water networks. Water Res. 164, 114926. https://doi.org/10.1016/j.watres.2019.114926.

Laub, P.J., Taimre, T., Pollett, P.K., 2015. Hawkes Processes.

Lewis, P.A.W., Shedler, G.S., 1979. Simulation of nonhomogeneous poisson processes by thinning. Naval Research Logistics 26, 403–413. https://doi.org/10.1002/nav.3800260304.

Le Gat, Y., 2014. Extending the Yule process to model recurrent pipe failures in water supply networks. Urban Water Journal 11, 617–630. https://doi.org/10.1080/1573062X.2013.783088.

Lin, P., Yuan, X.-X., 2019. A two-time-scale point process model of water main breaks for infrastructure asset management. Water Res. 150, 296–309. https://doi.org/10.1016/j.watres.2018.11.066.

Pritchard, O.G., Hallett, S.H., Farewell, T.S., 2014. Soil impacts on UK infrastructure: current and future climate. Proceedings of the Institution of Civil Engineers - Engineering Sustainability 167, 170–184. https://doi.org/10.1680/ensu.13.00035.

Rajani, B., Kleiner, Y., 2001. Comprehensive review of structural deterioration of water mains: physically based models. Urban Water 3, 151–164. https://doi.org/10.1016/S1462-0758(01)00032-2.

Redfern, T.W., Macdonald, N., Kjeldsen, T.R., Miller, J.D., Reynard, N., 2016. Current understanding of hydrological processes on common urban surfaces. Progress in Physical Geography 40, 699–713. https://doi.org/10.1177/0309133316652819.

Reinhart, A., Greenhouse, J., 2018. Self-exciting point processes with spatial covariates: modelling the dynamics of crime. J. Royal Statistical Soc. C 67, 1305–1329. https://doi.org/10.1111/rssc.12277.

Reinhart, A., 2018. A Review of Self-Exciting Spatio-Temporal Point Processes and Their Applications. Stat. Sci. 33, 299–318. https://doi.org/10.1214/17-STS629.

Scheidegger, A., Leitão, J.P., Scholten, L., 2015. Statistical failure models for water distribution pipes - A review from a unified perspective. Water Res. 83, 237–247. https://doi.org/10.1016/j.watres.2015.06.027.

Seica, M.V., Packer, J.A., 2004. Mechanical properties and strength of aged cast iron water pipes. J. Mater. Civ. Eng. 16, 69–77. https://doi.org/10.1061/(ASCE)0899-1561(2004)16:1(69).

Task Committee on Water Pipeline Condition Assessment, 2017. Pipe materials: asbestos cement, in: Ruchti, G.F. (Ed.), Water Pipeline Condition Assessment. American Society of Civil Engineers, Reston, VA, pp. 89–101. https://doi.org/10.1061/9780784414750.ch09.

Virtanen, P., Gommers, R., Oliphant, T.E., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E., Peterson, P., Weckesser, W., Bright, J., van der Walt, S.J., Brett, M., Wilson, J., Millman, K.J., Mayorov, N., Nelson, A.R.J., Jones, E., Kern, R., Larson, E., Carey, C.J., Polat, İ., Feng, Y., Moore, E.W., VanderPlas, J., Laxalde, D., Perktold, J., Cimrman, R., Henriksen, I., Quintero, E.A., Harris, C.R., Archibald, A.M., Ribeiro, A.H., Pedregosa, F., van Mulbregt, P., SciPy 1.0 Contributors, 2020. SciPy 1.0: fundamental algorithms for scientific computing in Python. Nat. Methods 17, 261–272. https://doi.org/10.1038/s41592-019-0686-2.

Zhuang, J., Ogata, Y., Vere-Jones, D., 2004. Analyzing earthquake clustering features by using stochastic reconstruction. J. Geophys. Res. 109. https://doi.org/10.1029/2003JB002879.

Zhuang, J., Touati, S., 2019. Stochastic simulation of earthquake catalogs, Community Online Resource for Statistical Seismicity Analysis. Corssa.