10-19-2019

Data-driven predictive maintenance scheduling policies for railways

Pedro Cesar Lopes Gerum
Cleveland State University, p.lopesgerum@csuohio.edu

Ayca Altay

Melike Baykal-Gürsoy

Recommended Citation

Data-driven Predictive Maintenance Scheduling Policies for Railways

Pedro Cesar Lopes Gerum, Ayca Altay, Melike Baykal-Gürsoy

Rutgers University, 96 Frelinghuysen Road, CoRE Building, Room 201, Piscataway, NJ 08854
pedro.gerum@rutgers.edu, ayca.altay@rutgers.edu, gursoy@soe.rutgers.edu

---

Abstract

Inspection and maintenance activities are essential to preserving safety and cost-effectiveness in railways. However, the stochastic nature of railway defect occurrence is usually ignored in literature; instead, defect stochasticity is considered independently of maintenance scheduling. This study presents a new approach to predict rail and geometry defects that relies on easy-to-obtain data and integrates prediction with inspection and maintenance scheduling activities. In the proposed approach, a novel use of risk-averse and hybrid prediction methodology controls the underestimation of defects. Then, a discounted Markov decision process model utilizes these predictions to determine optimal inspection and maintenance scheduling policies. Furthermore, in the presence of capacity constraints, Whittle indices via the multi-armed restless bandit formulation dynamically provide the optimal policies using the updated transition kernels. Results indicate a high accuracy rate in prediction and effective long-term scheduling policies that are adaptable to changing conditions.

Keywords: Rail Defect Prediction, Random Forests, Recurrent Neural Networks, Markov Decision Processes, Restless Bandits, Whittle Index

---

1. Introduction

The demand for rail transport is experiencing a boost at a global level, subsequently straining railway companies' ability to maintain service quality (Sharma et al., 2018). Simson et al. (2000) and Budai-Balke et al. (2009) argue that track maintenance and renewal costs are one of the major costs for railway companies. Corrective maintenances are spot-on inspections and repairs of existing defects, while preventive maintenances are inspections

and adjustments on defect-prone tracks (Sharma et al., 2018). Without effective inspection and maintenance, the safety and security of the public and goods utilizing rail networks cannot be ensured. However, inspections and maintenance activities may become costly if done excessively. Therefore, there is a need for efficient inspection and maintenance policies.

In rail terminology, geometry defects are horizontal and/or vertical misalignments in the track (Sharma et al., 2018), whereas rail defects include track wear such as corrosion or impairments such as broken rails or cracks (Clark, 2004). Some prior work classifies defects as yellow or red, depending on their severity (He et al., 2015; Cárdenas-Gallo et al., 2017), which will also be employed in this study. Yellow defects are minor defects, such as superficial cracks or buckling, and satisfy the Federal Railroad Administration (FRA) standards. On the other hand, red defects are major defects (e.g., broken rails) that do not satisfy FRA standards and need immediate repair.

Railway companies routinely face the problem of how to decide which segments to inspect and maintain. Such decisions depend on the number of defects a track is expected to have, making accurate defect predictions an important step in this process. In literature, the prediction of defects and the scheduling of maintenance activities are mostly treated as different problems due to their complex nature involving numerous constraints. Furthermore, literature avoids treating all possible defect types together due to the increase in complexity. A few studies (Sharma et al., 2018; Merrick and Soyer, 2015) integrate prediction and maintenance. Sharma et al. (2018) introduce a Markovian model to predict whether defects happen or not, and use Markov decision theory to determine the optimal inspection and repair policy. Merrick and Soyer (2015) employ a nonhomogeneous Poisson process to model the stochasticity of track failures in order to plan for their replacement. In this paper, we propose an innovative integration of defect prediction and optimal scheduling, as well as solutions for issues both in defect predictions, and inspection and maintenance scheduling. First, we account for the stochastic nature of defect prediction. Then, using the results obtained, we develop an iterative approach to schedule inspections with or without crew limitations optimally. Figure 1 visually describes the integration process that is used in this paper.

The remainder of thes paper is organized as follows: Section 2 reviews previous research.

Section 3 presents the defect and inspection data. Rail segments may have different defect characteristics. Hence, to be able to address the optimal inspection and maintenance scheduling problem in an efficient way, they have to be clustered into similar groups as described in Section 4. Section 5 details the prediction of defects using the available data, and section 6 describes how to use the prediction for unconstrained scheduling. Section 7 generalizes the scheduling of inspections and maintenance for a capacity-constrained problem and provides a framework for how to continuously improve the model with the addition of new data. In each of these sections, we have used the data presented in Section 3 extensively to provide examples. Finally, conclusions are presented in Section 8.
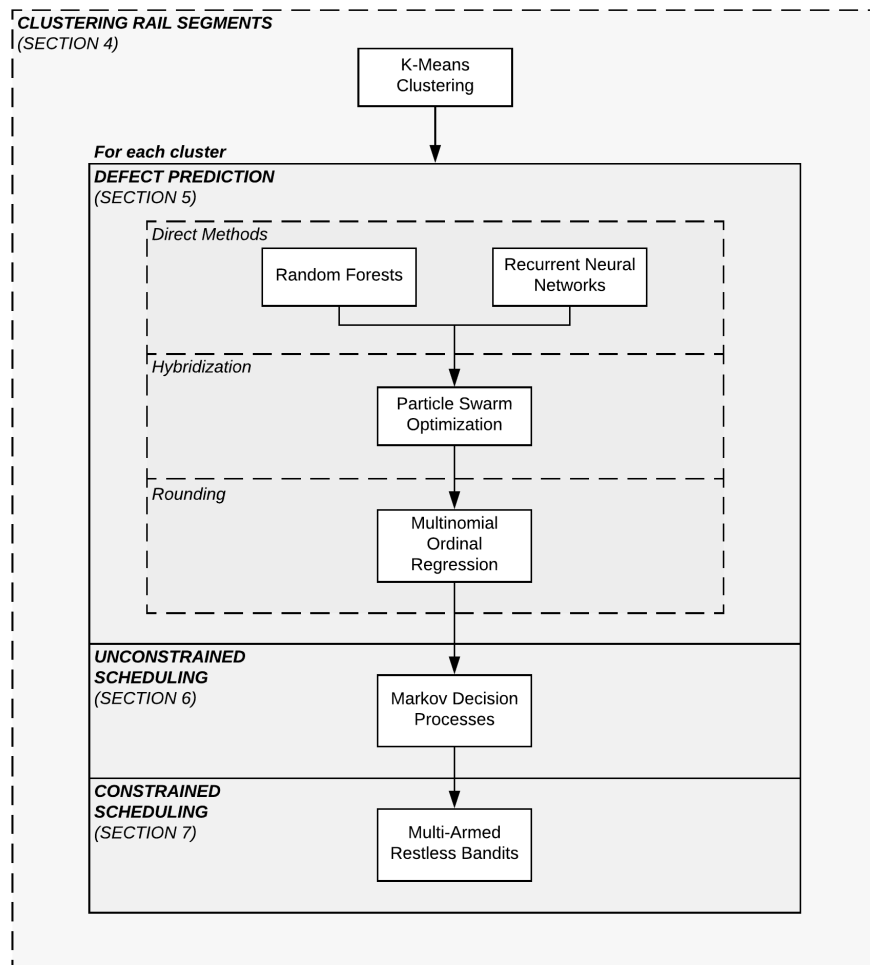

Figure 1: Flowchart of how defect prediction, and inspection and maintenance scheduling can be integrated.

2. Background

This section reviews first the previous work on defect prediction and then, on inspection and maintenance scheduling.

2.1. Defect Prediction

Earlier research employs summary statistics, especially standard deviation, to evaluate the risk of defects or derailments (Hamid and Gross, 1981). However, these summary statistics are not static over time and depend on conditions such as season, and load.

Other studies in defect prediction use classification and regression-based data analysis methods. Focusing on geometry defects, Sharma et al. (2018) develop a Track Quality Index (TQI) based Markov model. The authors divide rail tracks into 0.1-mile segments and compare Random Forests (RF), Logistic Regression (LR) and Support Vector Machines (SVM) models that predict a binary, defect/no defect, outcome. Martey et al. (2017) propose to predict geometry defects in a big data environment. Since analyzing big data is expensive, the authors cluster tracks and employ Principle Component Analysis (PCA) to identify the most important features that affect defect occurrence. In both studies, RF outperforms other methods.

Moridpour et al. (2017) introduce a regression-based model for the degradation level of light-rail tracks using Artificial Neural Networks. A similar study is conducted by Güler (2014) to predict degradation due to geometry defects using ANN. While this study prioritizes geographic and hardware-related inputs, Moridpour et al. (2017)'s study relies on railway traffic data. Both studies satisfy the requirement for detailed data for accurate prediction and declare a success rate of more than 70% for ANNs.

2.2. Inspection and Maintenance Scheduling Optimization

The scheduling of inspections and maintenances has been described as a complex, multivariate problem of major importance (Fan et al., 2011; Chen et al., 2014; Peralta et al., 2018; D'Ariano et al., 2019; Ghofrani et al., 2018). Previous studies attempt to solve this problem by implementing various adaptations of well-known optimization problems such as the Vehicle Routing Problem (VRP) or the Traveling Salesman Problem (TSP). However,

a common issue faced by most studies is the assumption that defects are deterministically known. Most authors fail to account for the stochasticity of where and when defects occur, and thus disregard the prediction problem.

Heinicke et al. (2015) characterize maintenance scheduling as a multi-depot VRP with time windows. Fan et al. (2011) also consider a VRP, but only include transportation (travel) costs. In both studies (Fan et al., 2011; Heinicke et al., 2015), however, the problems end up being NP-hard, thus making a polynomial run time to reach the optimal solution difficult to achieve.

Camci (2014) coins the term Traveling Maintainer Problem that corresponds to scheduling TSP with repair and inspections costs also included in the overall cost. Pour et al. (2018) construct a Mixed Integer Programming problem to assign teams according to their capabilities. Luan et al. (2017) solve this problem using a Lagrangian relaxation approach. A more extensive study is conducted by Lidén and Joborn (2017) further incorporates the railway traffic into inspection and maintenance scheduling. Nonetheless, all of these studies mention that the large number of variables results in considerable execution time due to the NP-hardness of the mathematical programs. Hence, heuristics and metaheuristics are frequently implemented to obtain a feasible schedule (Camci, 2014; Andrade and Teixeira, 2016; Khalouli et al., 2016). Column generation methods are employed to find exact optimal solutions for large-sized mathematical programs (Nishi et al., 2011; Lannez et al., 2015).

Furthermore, there are some studies on risk-based approaches that incorporate qualitative data into the maintenance scheduling problem. Wang et al. (2018) break down failure consequences into fuzzy linguistic classes from negligible to catastrophic, and qualify failure likelihood from very low to very high. Consilvio et al. (2016) consider the problem of how to choose threshold levels to achieve a tolerable degradation range. A similar risk-based approach by Jiang et al. (2003) employs multiple degradation states with associated probabilities of failure to decide on the maintenance schedule.

In a recent review of railway transportation, Ghofrani et al. (2018) classify railway maintenance as: i) condition-based, ii) preventive, and iii) corrective maintenance. The subject matter of this paper, predictive maintenance, is accounted as a special topic within condition-based maintenance. According to Ghofrani et al. (2018), literature mostly deals with track

defects using corrective maintenance, and scheduling is mostly planned in cases when defects are already known. Moreover, condition-based maintenance is mostly exploited for vehicle maintenance but rarely applied to tracks. Turner et al. (2016) provide an extensive literature review on planning and scheduling of railway traffic in Europe, also discussing a few studies that incorporate maintenance activities into transportation planning and scheduling.

Next, we present the data that will be later used in the examples.

3. Preprocessing Rail Data

Most rail companies gather data to understand how their processes behave, including defects and inspections. However, due to human interaction in data registration, as well as the lack of enforced standards, many companies may hardly maintain the accuracy of such data. Furthermore, some companies also struggle with the availability of detailed data for prediction and scheduling. Therefore, there is a trade-off between the benefits of process optimization through data analyses, and the cost of obtaining accurate and detailed data. In this section, we explain faulty data treatment and noise filtering while maintaining an acceptable level of data.

In this study, we use rail data that was gathered over a two-year period (2016 and 2017). The defect database includes a list of all rail and geometry defects found in each segment of the network, detailing the time and the location in which the defect was found, and the type and severity (yellow or red) of the defect found. The inspection data include the type of inspection, the segment inspected, and the time of inspection. However, the inspection data do not provide the exact beginning and end of inspection locations. Moreover, segments vary from 300m to 60km in length, with an average of 17 km. Figure 2 displays spatio-temporal defect observations on a particular segment. The horizontal axis represents the whole length of the rail segment while the vertical axis represents time. The blue dots on the vertical axis correspond to the inspections. Finally, a third database contains the daily load information for each segment. Load information is determined as the total gross tonnage endured by the segment, including the weight of wagons. The number of wagons transported or wheels in contact with the track is not available.

The defect data do not hold records for the related inspection and vice versa. To better
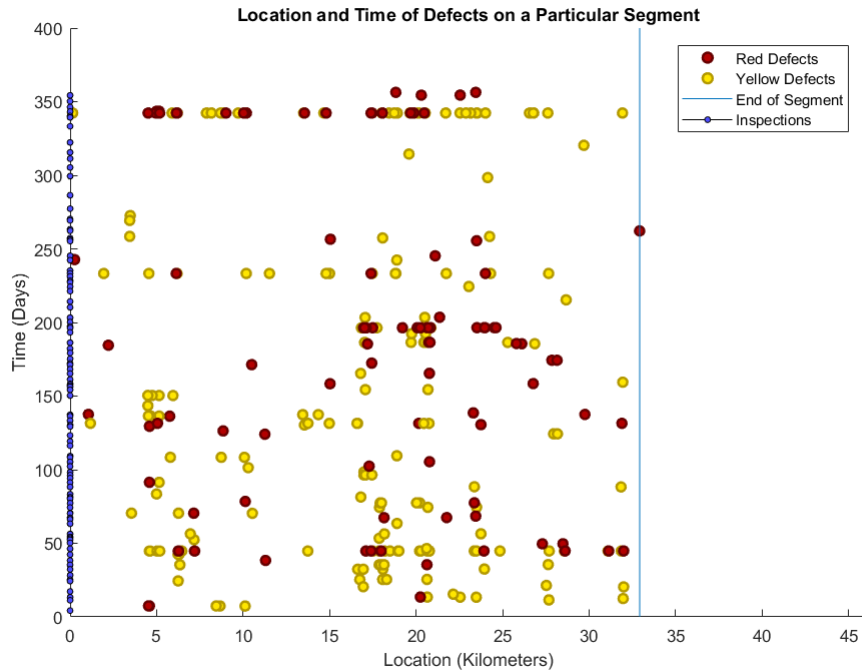
Figure 2: Spatiotemporal defect observations.

assess how the number of defects per inspection is distributed, each defect (rail or geometry) found in the defect dataset is matched to a corresponding inspection. This matching is one-to-many; a defect can be found during only one inspection, an inspection can find multiple defects. Through this process, we are able to achieve 91% match, losing only 9% of the defect data. For each inspection, we match the defects that are located in the inspected segment, found through the specific inspection type, and registered during the time of the inspection with a buffer of no more than one day added to represent the delay between the time the inspection outcomes are recorded and the inspection execution time. Unmatched data may represent a human error, as well as data systems' mismatches, so they are removed. Faulty data, such as inspections with negative execution times, or with negative inter-arrival times have also been also removed, along with the defects that match them.

The raw data contains more than 26 000 inspections and 82 000 defects. Broken rails are one of the most commonly encountered red defect types (Figure 3). Moreover, they are most often rated as red: around 96% of broken rails are classified as such. Table 1 shows the defect types that are most commonly rated as red. Note that the visual and ultrasound
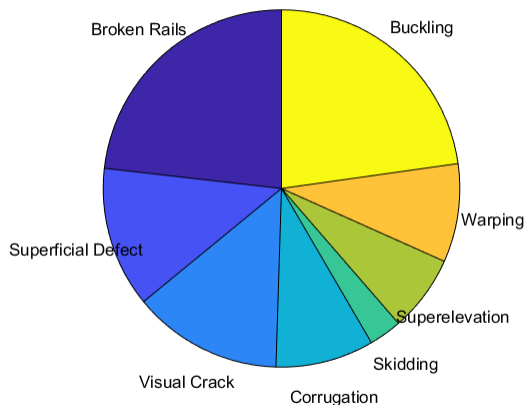
cracks usually precede broken rails.

Figure 3: Distribution of defects classified as 'red' per type of defect.

Defect Type	Percentage of Defects from each type classified as 'red'
Broken rails	95.97%
Visual Cracks	58.05%
Ultrasound cracks	52.52%
Warping	39.88%
Buckling	32.25%

Table 1: Defect types with highest rates of red defects.

4. Clustering Rail Segments

This problem involves a network that may contain numerous rail segments. These segments differ in terms of their defect-proneness due to the external conditions they may endure: different loads, seasonal conditions, etc. While a generalized prediction method may fail to capture defect characteristics of all segments, implementing a prediction method for each rail segment is inefficient and not reliable due to the lack of data. Clustering similar segments pools sufficient data for prediction (Pour and Benlic, 2017; Doric et al., 2017).

Feature selection is vital as every incorrect feature might increase the rate of misclassified data (Azizyan et al., 2014). In this study, the feature space includes 14 features extracted through the available data for 170 rail segments. Four defect groups considered are: all defects, red defects, yellow defects, and broken rails. For all defect groups, three statistics are recorded: the annual number of defects, the average time between defects (in days), and the average number of defects found per inspection. Two additional features selected are the length of the segment and the sustained load. In order to find the best representation, each subset of features is enumerated. Each subset is then fed into the K-Means algorithm (Khan and Ahmad, 2004) in order to make two decisions: the features selection, and the selection of the optimal number of clusters. The validity of each clustering setting is measured by the C-index defined by Hubert and Levin (1976) as

$$C = \frac{S - S_{min}}{S_{max} - S_{min}}, \tag{1}$$

where S is the sum of pairwise distances among all data points that lie in the same cluster, S_{min} is the sum of the smallest n pairwise distances among all data points, and S_{max} is the sum of the largest n pairwise distances among all data points. A smaller index value indicates better clustering.

The lower and upper bounds for the number of clusters are set in a way that satisfies the following properties: (i) sufficient data remains for prediction in each cluster, (ii) the ratio of the outlier rail segments to all rail segments is not large, and (iii) the number of clusters that contain each outlier segment is not large. The algorithm is run 10 times for each possible number of clusters for a higher accuracy to negate the effect of random initial cluster centers (Khan and Ahmad, 2004). The cluster number with the smallest average C-Index of 10 runs is selected. Even with small C-Index values, some undesirable outcomes, such as high misclassification rate or clustering imbalance may arise. A threshold for misclassification rate is chosen as 10% (Bioch and Popova, 2002), and a limit of 60% of segments is set to ensure the balance in clustering.

Results

The three features selected by the algorithm with best clustering results are (C-index: 0.0894): *i*) the annual number of broken rails, *ii*) the average time in days between broken

rails, and *iii*) the average number of all types of defects per inspection. Two out of three features involve broken rails, which is the most common defects in all rail segments. These attributes determine the similarity of defect characteristics of rail segments; however, the emergence of defects may depend on other criteria. It is the task of the prediction module to reveal possible input combinations that lead to similar values of these three defect characteristics. Although the lengths of segments vary, the inspection length tends to be consistent, thus offsetting a possible bias. Moreover, the dataset suggests no correlation between the length of a particular segment and the number of inspections performed on that segment (see Figure 4).
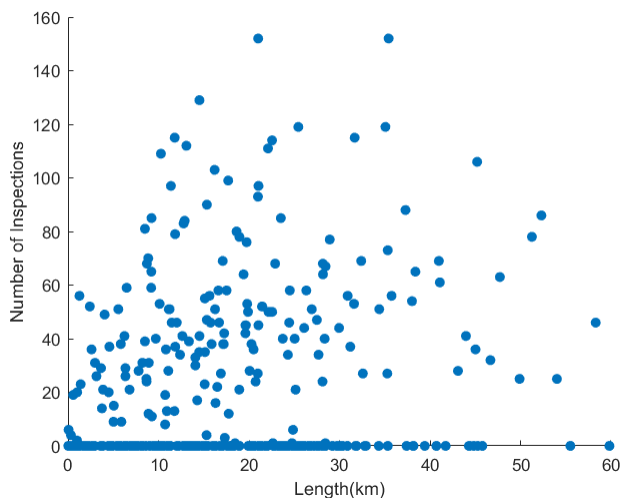

Figure 4: This figure shows the lack of correlation between the number of inspections performed and the length of the segments.

Once the feature selection is completed, the number of clusters is then decided. Figure 5 displays the average C-index with respect to the number of clusters using these three attributes. In literature, there are other algorithms that do not require an initial setting for the number of clusters such as agglomerative hierarchical clustering. However, rail clustering has challenges that may not exist in other situations: for example, *i*) the linkage parameter has to be manually adjusted in a way that gives a clear cutoff point for clustering, *ii*) even with best cutoff points, some clusters do not have the best possible C-index, *iii*) for some of the subsets, even the best cutoff point leads to the violation of the clustering constraints presented above (i.e., large number of outliers, small and excessive number of clusters).

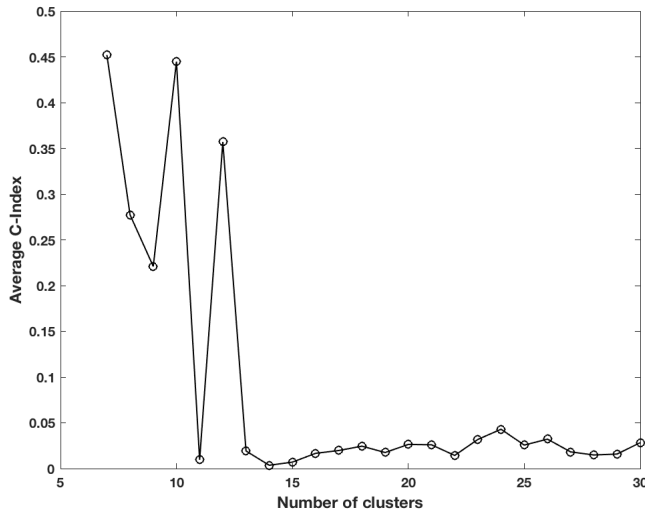K-Means provides sufficient flexibility to avoid these challenges.

Figure 5: Number of cluster vs. C-index for 170 rail segments.

Figure 5 indicates that the minimum average C-Index is obtained in the case of 14 clusters (C-index: 0.0036), out of which 4 contain a total of 140 rail segments (82.8% of all rail segments), and the remaining 10 contain only a few rail segments with insufficient data. More specifically, clusters 1 to 4 contain 24, 28, 47 and 41 rail segments with sufficient data for prediction; hence, the predictions and the scheduling model will be evaluated over these 140 rail segments using 4 clusters.

5. Non-linear Regression Models for Defect Prediction

Literature agrees that predicting the number of defects in a track is mostly a non-linear problem. Most authors aim for closed-form equations for prediction but advanced 'black-box' techniques are replacing them because these techniques enable the employment of different feature sets as inputs. Machine-learning approaches, such as RF and ANNs have recently been used for defect prediction (Güler, 2014; Moridpour et al., 2017; Sharma et al., 2018), but very little has been done on the integration of such prediction models with inspection and maintenance scheduling policies. Furthermore, many studies use features that are often not available to rail companies. In this study, we focus on methods that use data that are widely available in most locations. In sections 6 and 7, we integrate these approaches into

11

the scheduling model to increase the efficiency of inspection and maintenance scheduling policies. In this section, we compare two common methods to predict the number of yellow and red defects: Random Forests and Recurrent Neural Networks (Sharma et al., 2018).

Random Forest (RF) is an ensemble regression model that combines independent regressors, namely decision trees. Each decision tree is constructed by randomly selecting a subset of features from the set of all features, and by using the bootstrap aggregation technique. The prediction is determined by averaging the outputs of all decision trees. The use of multiple trees reduces the chance of over-fitting, and decreases the variance (Friedman et al., 2001). Each tree is generated by sampling the training data with replacement, and by using a randomly generated set of 5 features from the 7 possible features. Then, trees are grown with the criterion of using the maximum impurity gain from all candidates to split branches. The impurity of each node is calculated with the Gini's diversity index to determine the impurity of each node (Breiman et al., 1984). We grow 300 trees in this modeling approach.

Recurrent Neural Networks (RNNs) are special types of Neural Networks that recursively calibrate the prediction accuracy by providing the lagged output as an additional input (Burden and Winkler, 2008). RNNs produce different results each time they are run, and thus, 10 networks are trained with the same data, and their results are averaged in order to obtain the final result. The algorithm parameters are 10 neurons on one hidden layer, and α and β parameters coming from a uniform distribution (0,10). A deeper network is not chosen in order to avoid overfitting and high computation time. RNNs are known to perform well for large databases and the RNN in this study does not take inputs for rail segments directly; its inputs are the integrated defect and inspection data. Hence, even though there are 170 segments overall, all segments in 4 clusters have a sufficiently large number of inspections to render reasonable confidence level in the test results.

The prediction model is run separately for red defects, and yellow defects, which are more numerous than yellow defects. The inputs of both models, as mentioned above, involve the number of red and yellow defects the previous inspection has found; hence, the inputs are inspection related. The number of inspections is the same for both red and yellow defects; any inspection can find both types. Thus, red and yellow defect prediction models have the same number of inputs. The prediction model determines different structures for the

relationship between the inputs and the outputs for red or yellow defects.

Finally, since the predictions results are real numbers, we use Multinomial Ordinal Regression (MRO) to assign each prediction to its corresponding number of defects.

5.1. Results

In this section, we provide the results of using RF and RNN to predict rail defects of different severity levels. We use the standard loss function that maximizes accuracy by analyzing mean absolute error and mean squared error. However, some risk-averse decision-makers may consider false negatives worse than false positives. For such cases, we propose a new formulation in Section 5.2.

Both methods exhibit similar results and show a small error rate that matches the one in literature, even though fewer and more commonly found features are used. Sharma et al. (2018) have recently predicted the existence of a geometry defect at an accuracy rate of 75-77%. Cárdenas-Gallo et al. (2017) apply the red/yellow distinction to railway defects and their accuracy is also around 80%. Performance measures obtained by RNN also closely match those from the RF model, indicating that either model can be applied. Due to the high likelihood of inspections finding few to no defects, the models tend to undershoot more often than overshoot. RNN has slightly higher error rates for both yellow and red defects and has a slightly higher undershooting rate than RF.

Tables 2a and 2b display the prediction of defects, red and yellow, respectively, during a walking inspection (0, 1, or 2 or more defects). Predicting a more granular number of defects demands more data points than are available in this study, but an increase in granularity is recommended as more data become available. 80% of the data is used to train the model, with the remaining 20% used for testing. The third and fourth columns describe the expected over and under-predicted number of defects, respectively. The average number of over-predicted defects is significantly skewed by extremely infrequent data points with no defect when it is predicted to have two defects. Note that over-prediction of defects occurs in less than 2% of all inspections, except in one cluster, where it occurs in around 4% of inspections.

Figures 6a and 6b depict the average accuracy for all clusters during one run. The overall accuracy of perfect (spot-on) prediction for testing data is around 82% for red defects and

Cluster	MAE	MSE	E[over]	E[under]
1	0.31	0.48	2.00	1.38
2	0.32	0.49	-	1.36
3	0.26	0.44	1.57	1.50
4	0.25	0.41	2.00	1.47

(a) Red Defects - RNN Error Results. Mean Absolute Error and Mean Squared Error in number of defects per inspection. The round numbers on the expected overestimation are caused by a single overestimates.

| Cluster	MAE	MSE	E[over]	E[under]
1	0.40	0.59	1.50	1.28
2	0.51	0.82	1.44	1.44
3	0.53	0.77	1.31	1.31
4	0.41	0.62	1.88	1.28

(b) Yellow Defects - Random Forest Error Results. Mean Absolute Error and Mean Squared Error in number of defects per inspection. The round numbers on the expected overestimation are caused by a few overestimates.

Table 2: Prediction results using RF and RNN.

62% for yellow defects on average. Yellow defects are harder to predict due to their high variance in the data. Given the risk-averse characteristics of the problem, it may be preferable to over-predict rather than to under-predict if one considers over-predictions acceptable.
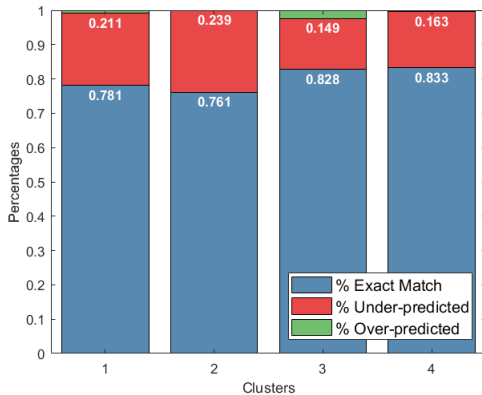
(a) Red Defects - RNN Prediction Results.
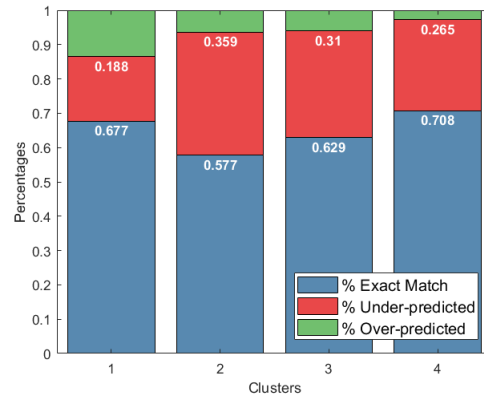
(b) Yellow Defects - RF Prediction Results.

Figure 6: Underestimation, overestimation and exact prediction rates.

5.2. Risk-averse adaptation

In designing systems for critical infrastructures, reducing missed observations is more important than removing unnecessary screenings (Thomas et al., 2001). For railway systems, a missed observation refers to an overlooked defect, whereas an unnecessary screening refers to an inspection of a defect-free rail segment. The risk of derailment and its possible consequences may be more costly than performing an extra inspection. Hence, weights for

underestimations may be strictly larger than for overestimations.

Results obtained by both RF and RNN tend to favor undershooting due to the class imbalance of the data. The logistic regression results provide the confidence level of the model for each possible prediction option. The standard selection chooses the prediction with the highest confidence level, ensuring a higher proportion of exact matches.

A proper means to account for the proportion of undershooting is to hybridize the two models with the objective of decreasing the rate of underestimation while maintaining a reasonable level of overestimation. In this setting, the loss function below reflects the importance given to the underestimation,

$$
\epsilon(r) = \begin{cases} r, & \text{if } r \leq 1 \\ \frac{1}{2}(r^2 + 1), & \text{if } r > 1, \end{cases}
$$

where $r = y - \hat{y}$, y is the actual number of defects and $\hat{y}$ is the predicted number of defects. This loss function is continuous, differentiable and has a minimum at $r = 0$. However, the predicted value is piecewise linear, that is,

$$
\hat{y} = [a + b_0 \hat{y}_{RF} + b_1 \hat{y}_{RNN}], \tag{2}
$$

where $[\cdot]$ denotes the rounding process using MRO. Hence, the minimization function becomes

$$
\min z = \sum_{i, r_i \leq 1} r_i^2 + \sum_{i, r_i > 1} [\frac{1}{2}(r_i^2 + 1)]^2, \tag{3}
$$

where $r_i = y_i - \hat{y}_i$, with i as the index of the data point.

Furthermore, the objective function 3 relies on the square of negative errors to amplify their impact and forces the policy to shift from undershooting to overshooting. The values of a, b_0 and b_1 are selected to minimizes the squared error. This objective function is convex and differentiable as long as the rounding process in MRO is ignored. However, due to rounding, a closed-form objective function cannot be written and traditional nonlinear optimization techniques cannot be implemented. Hence, an algorithm that does not require a closed-form performance measure, Particle Swarm Optimization (PSO) algorithm is selected to predict the values of a, b_0 and b_1 (Eberhart and Kennedy, 1995). Its parameters are determined as the number of particles being equal to 1000, the inertia coefficient being set as 0.2, and the

cognitive and social coefficients being set as 1. The algorithm is run 20 times for each cluster and defect severity, and for each run, the resulting local minima are recorded. The best local optimum value is accepted as the solution.

Figures 7, 8, and 9 illustrate, for each cluster, the percentage of underestimated data, the rate of false negatives, that is, the percentage of zero defect predictions when there is at least one defect, and the average underestimation, respectively. The figures compare the results among RF, RNN, and the hybrid method.
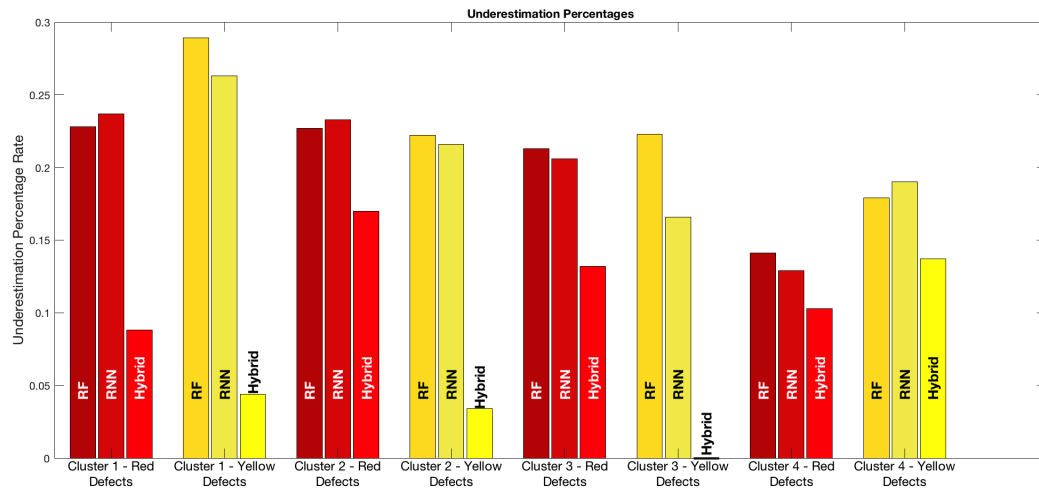

Figure 7: Percentage of underestimated data for each of the three approaches for each cluster.
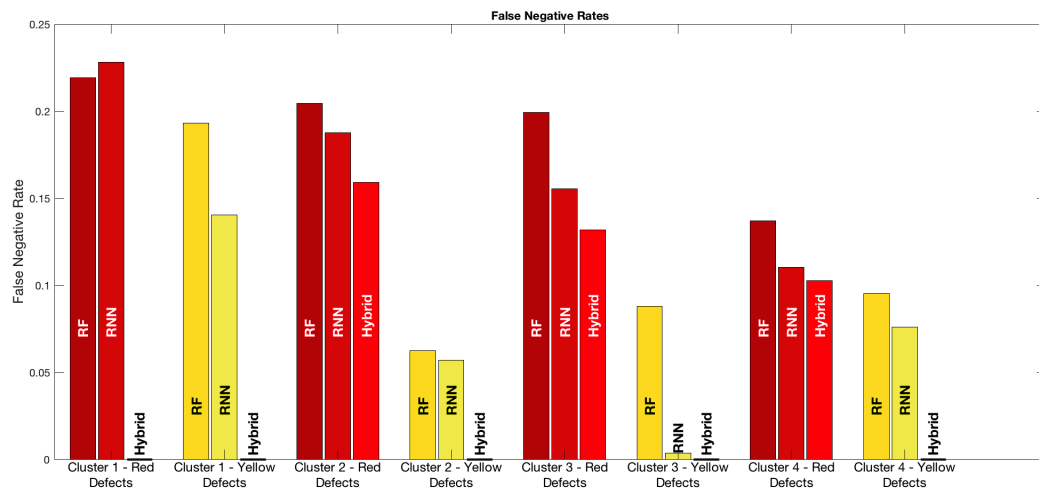

Figure 8: Rate of false negatives for each of the three approaches for each cluster.
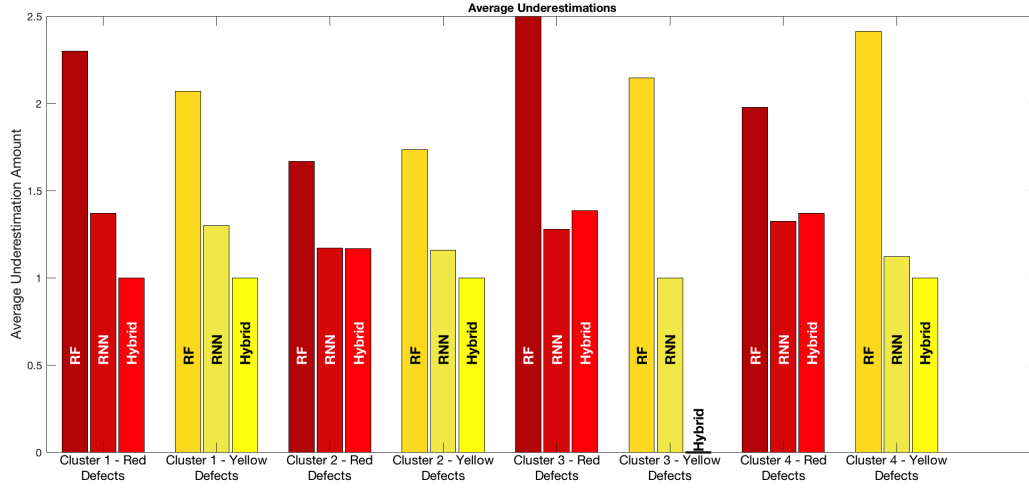
Figure 9: Average underestimation for each of the three approaches for each cluster.

The rate of underestimation and the rate of false negatives are significantly reduced by the hybridization. The overall average false negative rate is decreased to less than 5% from 15%, whereas the overall average undershooting percentage is decreased to 8% from 20%. As for the expected underestimation, there are two cases in which the hybrid method performs worse than pure methods, yet the average underestimation is decreased from 1.30 to 1.00. Other performance measures are presented in tables 3a and 3b for red and yellow defects, respectively. Given the model's trade-off between overestimations and underestimations, the overall expected overestimation has been increased from 1.00 to 1.50. However, this increase can be considered acceptable in view of the decrease in underestimation.

Cluster	MAE	MSE	E[over]
1	0.877	0.930	1.034
2	0.438	0.619	1.355
3	0.733	1.334	1.831
4	0.357	0.631	1.839

(a) Performance Measures of the Hybridization for Red Defects.

Cluster	MAE	MSE	E[over]
1	1.228	2.053	1.534
2	1.330	2.352	1.652
3	1.385	2.486	1.660
4	0.810	0.819	1.000

(b) Performance Measures of the Hybridization for Yellow Defects.

Table 3: Performance Measures for the Hybrid Model.

6. Scheduling Inspection/Maintenance

Most authors assume a known list of defects across the network before scheduling maintenance activities. However, when this information is not known with certainty, one has to resort to stochastic methods to find the optimum inspection policies. In this section, we propose a Markov decision process (MDP) model (Ross, 1992; Puterman, 1994; Bertsekas, 2007) that integrates the stochastic nature of defect occurrence into scheduling. Through this model, one can determine the optimal inspection policy for a specific segment over an infinite time horizon. An assumption used is that all inspections are performed at the beginning of the day, and all defects observed during the inspection are repaired instantaneously. New defects might arise during the day after the inspection. Hence, one can model the segment deterioration state as a discrete time Markov process that depends on the previous state and the action taken in that state. Then, the Markovian decision problem to determine the sequence of actions that minimize the discounted cost incurred over an infinite horizon becomes

$$\nu(s(0)) = \min_{a(t) \in \mathcal{A}} E\left[\sum_{t=0}^{\infty} \alpha^t c(s(t), a(t)) \,|\, s(0)\right], \tag{4}$$

where $\nu(\cdot)$ denotes the optimal total discounted cost, i.e., value function, starting in state $(\cdot)$, $s(t) \in \mathcal{S}$ denotes the segment state at time t with $\mathcal{S}$ representing the set of states, $a(t) \in \mathcal{A}$ denotes the binary variable representing if the segment is inspected or not at time t with $\mathcal{A}$ representing the set of actions, and $c(\cdot, \cdot)$ denotes the expected cost incurred when the state and action information are given. α represents the discount factor, and the notation $E[\cdot|s(0)]$ represents the conditional expectation given the initial deterioration state of the segment. Later on, we generalize our approach to the case of multiple segments under a constraint on the availability of inspection teams.

Next, we discuss the dynamic programming formulation of the problem and the details of the MDP model.

6.1. Dynamic Programming Formulation

A direct method to generate the optimal policy for equation 4 is to use the following dynamic programming equation (Ross, 1992; Puterman, 1994),

$$\nu(s) = \min_{a \in \mathcal{A}} \left\{ c(s, a) + \sum_{s' \in \mathcal{S}} \alpha \nu(s') \cdot p\{s'|s, a\} \right\}, \forall s \in \mathcal{S}, \tag{5}$$

with p denoting the transition kernel, and $\nu(s)$ representing the expected discounted cost to go starting in state s from the current period onwards. Let $c(s, a)$ correspond to the expected cost of inspection and repair if the action is 'inspect', and the expected cost of the risk of derailment otherwise. Since the state space, $\mathcal{S}$, and the action space, $\mathcal{A}$, are finite, $c(s, a)$ is uniformly bounded. $p\{s'|s, a\}$ denotes the probability of moving to a new state s', given the current state s and the action taken in the current period, a. These probabilities are estimated from the data. In this problem, we fix the discount rate as $\alpha = 0.95$.

Let us define the state of the segment as a composition of the current level of rail deterioration and the load during the next period. The rail deterioration level is 1, when no red or yellow defects are present; 2, when no red defect is present; and 3, when at least one red defect is present. Load is also separated into two states as high and low, where the cut-off is the median load endured by the whole network, obtained from data. There are, therefore, six possible states: $(1, L), (2, L), (3, L), (1, H), (2, H), (3, H)$, denoted as $1, 2, \ldots, 6$, respectively.

After filtering the inspection data by one day inter-inspection time within each cluster, the probability transition matrix for the action 'inspect', denotes as P_{insp}, is obtained from the frequency of each state change. These normalized frequencies correspond to the maximum likelihood estimators of the transition probabilities (Bartlett, 1951; Anderson and Goodman, 1957).

Two methods are used to validate the transition matrix given in Table 4a:

1. *Chi-Square testing for validation:* This test compares the candidate for the transition matrix to an independent trials matrix that is randomly generated using the transition frequencies (Hiscott, 1981). Let n be the number of states, F_{ij} be the number of transitions from state i to j, SR_i and SC_j be the ith row sum and j^{th} column sum of the F matrix, and T be the sum of all elements in the matrix. Then, the independent trials matrix (IT)

State	1	2	3	4	5	6
1	0.5266	0.1339	0.1072	0.1518	0.0447	0.0358
2	0.1772	0.2910	0.1899	0.2278	0.1013	0.0128
3	0.2058	0.1912	0.2793	0.1912	0.0148	0.1177
4	0.0981	0.0785	0.0719	0.4705	0.2026	0.0785
5	0.0556	0.1805	0.0695	0.3471	0.2916	0.0556
6	0.2352	0.0590	0.1471	0.3233	0.0590	0.1765

State	1	2	3	4	5	6
1	0.5266	0.1339	0.1072	0.1518	0.0447	0.0358
2	0.0000	0.6605	0.1072	0.0000	0.1965	0.0358
3	0.0000	0.0000	0.7677	0.0000	0.0000	0.2323
4	0.0981	0.0785	0.0719	0.4705	0.2026	0.0785
5	0.0000	0.1765	0.0719	0.0000	0.6731	0.0785
6	0.0000	0.0000	0.2485	0.0000	0.0000	0.7515

(a) Transition Matrix for the action 'inspect'.   (b) Transition Matrix for the action 'do not inspect'.

Table 4: Transition matrices for each action.

is written as

$$IT_{ij} = \frac{SC_j}{T - SC_i}.$$ (6)

The null hypothesis is that there is no significant difference between the random matrix and the predicted transition matrix. The χ^2 statistic is evaluated as

$$\chi^2_{observed} = \sum_{i=1}^{n} \sum_{j=1}^{n} \frac{(F_{ij} - SR_i \cdot IT_{ij})^2}{SR_i \cdot IT_{ij}},$$

and tested at $(n^2 - 2n)$ degrees of freedom. In the case of the 'inspect' transition matrix, $\chi^2_{observed}$ is calculated as 45.38 and the $\chi^2_{threshold}$ is 13.84. Therefore, there is enough evidence that the null hypothesis is rejected at a 0.05 significance level.

2. *Correlation with the two-step matrix*: A validation method used by Sharma et al. (2018) is based on the two-step state transition matrix, $P2$. If the one-step transition matrix is P, then the correlation between P^2 and $P2$ must be high. In our case, the correlation is found to be strong at level 0.8633.

Both methods validate the transition matrix for 'inspect', which is presented in Table 4a. We use this transition matrix as a starting point to determine the transition matrix for the action 'do not inspect'.

Let us look at the transition matrix for the action 'inspect' separately for high and low load (only considering the 3 deterioration states). The first row of the transition matrix for the action 'inspect' represents, given there are no defects found at the current inspection and the load is low, the probability of no red and no yellow defects happening, given as $\bar{P}_{red}$

and $\bar{P}_{yellow}$, respectively, the probability of no red defects happening and some yellow defects happening (P_{yellow}), and the probability of at least one red defect happening (P_{red}), given as $(\bar{P}_{red} \cdot \bar{P}_{yellow}, \bar{P}_{red} \cdot P_{yellow}, P_{red})$. We use these parameters to generate the transition matrix for the action 'do not inspect'. The other rows are similarly defined for the other initial states.

The transition probability for the action 'do not inspect' will be an upper triangular matrix, because no repairs are performed when no inspection happens. The first row is composed of $(\bar{P}_{red} \cdot \bar{P}_{yellow}, \bar{P}_{red} \cdot P_{yellow}, P_{red})$, which is the same as the transition matrix for the action 'inspect'. The second row is formed by $(0, \bar{P}_{red}, P_{red})$, because the system remains in state 2 if no red defects are generated, and moves to state 3 otherwise. Finally, the third row is formed as $(0, 0, 1)$. With the transition probability for the action 'do not inspect' determined for high and low load, we can rebuild the 6-state transition matrix through algebraic manipulations using the low load transition kernel, the high load transition kernel, and the load transition kernel. This matrix are shown in tables 4a and 4b. The matrices P_{insp} and P_{not} are based on the data and may not represent the "true" transition kernels. A structured approach on how to continuously update these matrices with the addition of incoming data, while still optimally choosing segments to inspect is presented in section 7.

When $a = 0$, $c(s, a)$ can be defined as the risk cost dependent on the number of defects existing at the beginning of the current period and the likelihood of new defects occurrence. This cost can be computed by multiplying the cost of a derailment by the probability that a derailment will occur given the number of defects present. Using Bayes' rule, the probability of a particular defect causing a derailment, $p\{\text{der.}|\text{def.}\}$, can be expressed as:

$$p\{\text{der.}|\text{def.}\} = \frac{p\{\text{def.}|\text{der.}\} \cdot p\{\text{der.}\}}{p\{\text{def.}\}}.$$

Given a derailment, the probability of its being caused by a certain defect, $p\{\text{def.}|\text{der.}\}$, is calculated by Liu et al. (2012). Comparing the data provided in Liu et al. (2012) with our data, we infer that the probability of a derailment being caused by a red defect is 22.6% and the probability of a derailment being caused by a yellow defect is 9.6%.

Furthermore, Anderson and Barkan (2005) derive the probability of a derailment, depending on the length of the train. On average, they report the probability of a train derailing

21

as $p\{\text{der.}\} = 0.720 \cdot 10^{-3}$. Finally, the probability of defect occurrence $p\{\text{def.}\}$ is given by P_{red} and P_{yellow} that are obtained from the transition matrix for the action 'inspect'.

Therefore,

$$p\{\text{der.}|\text{red defect}\} = \frac{22.6\% \cdot 0.720 \cdot 10^{-3}}{P_{red}},$$

$$p\{\text{der.}|\text{yellow defect}\} = \frac{9.6\% \cdot 0.720 \cdot 10^{-3}}{P_{yellow}}.$$

The cost of taking action 'do not inspect', $c(s,0)$, is just the probability of at least one of the defects causing a derailment. It can be obtained from the equation below, where c_{der} represents the expected derailment cost, and R and Y are random variables denoting the number of red and yellow defects present, respectively:

$$c(s,0) = c_{der} \cdot \sum_{r,y} \left[1 - \left((1 - p\{\text{der.}|\text{red def.}\})^r \cdot (1 - p\{\text{der.}|\text{yel. def.}\})^y \right) \right] p\{R = r, Y = y|s\}.$$

In this study, the joint distribution conditioned on the current state, $p\{R = r, Y = y|s\} \ \forall \ r, y$, is obtained empirically from the data. Finally, we multiply the cost by a factor of 1.6 for the states containing high load, accounting for the higher impact a derailment could bring, i.e. $c(i, H, 0) = 2c(i, L, 0)$ for $i = 1, 2, 3$.

We assume that other costs, such as damage to the image of the company, or lives lost are included in the cost of a derailment. Such costs can be adapted to account for different risk levels.

Furthermore, when $a = 1$, $c(s,a)$ includes the inspection and repair costs. For purposes of this research, we assume the inspection cost to be linearly dependent on the inspection length, and the repair cost to be an increasing function of the expected number of defects. Hence, it is also increasing in the state.

$$c(s,1) = E[\$ \text{ insp.}] \quad + \sum_{i=\{red,yel.\}} E[\# \text{ def.}_i|s] \cdot E[\$ \text{ rep.}/\text{def.}_i].$$

In this equation, the cost of inspection accounts for crew time, equipment used, and delays caused, and the cost of repair accounts for parts, manpower, and delay costs. The expected number is also obtained from the data. It should be noted that the parameters used in both cost functions can also be estimated from the regression models described in

Section 5. Each segment may have specific cost parameters, and should, therefore, have a segment-specific policy to optimize the scheduling as well.

The solution of this MDP model obtained via the dynamic programming equation provides the optimal action for every state. However, MDPs assume the knowledge of the current state to be able to forecast the future states. Consequently, the "state" definition should include all the information one needs to forecast the next state. In the case of segments that were not inspected during the previous period, the decision maker faces the problem of not being able to observe the current state.

In order to provide the full state information, we augment the state space by the number of days since last inspection (Bertsekas, 2005). If there was an inspection yesterday, the inspection revealed red defects, and the load observed was High, then the augmented state becomes $\mathbf{s} = (3, H, 0)$. We limit the time delay to at most 9 days with no inspection, since that is the maximum delay we have observed in the data. The new state space is, therefore, the set of all combinations of deterioration levels (1,2 or 3), load information (low or high), and days since the state was last observed (0-9), i.e., $\mathbf{s} = (i, j, k)$, for $i = 1, 2, 3$, $j = H$ or L, and $k = 0, 1, \ldots, 9$. The cardinality of the state space, $|\mathcal{S}|$, is $3 \cdot 2 \cdot 10 = 60$. Then, the new transition matrices are presented in tables 5a and 5b, where P_{insp} and P_{not} are the transition matrices displayed in tables 4a and 4b, respectively. Because we limit the maximum number of days since last inspection, P_{bound} is the boundary matrix with all states transitioning to 6 with probability 1, i.e. $([\mathbf{0}, \mathbf{0}, \mathbf{0}, \mathbf{0}, \mathbf{0}, \mathbf{1}]_{(6x6)})$, where $\mathbf{0}$ corresponds to a column vector of all zeros, and $\mathbf{1}$ corresponds to a column vector of all ones. It forces the system to change the action to inspect after at most 9 periods. However, most set of realistic parameters see an immediate switch. This structure with a finite set of states and P_{bound} as a boundary condition improves the robustness of the model.

The new cost vectors also need to be updated. The cost of the action 'do not inspect' for state s, is the same as before, but accounts for the expected next state. For instance, if the cost vector of not inspecting states $s \in \{1, 2, \ldots, 6\}$, where s contains the deterioration and load information is $\mathbf{c_{not}} = (c(1, 0), c(2, 0), \ldots, c(6, 0))^T$, the cost vector associated with $(\mathbf{s}, t)$, where t represents the time since the last inspection, is $P_{not}^t \cdot \mathbf{c_{not}}$. The extended cost of inspecting is constructed similarly across all states $(\mathbf{s}, t)$.

State	$(\cdot,\cdot,0)$	$(\cdot,\cdot,1)$	$(\cdot,\cdot,2)$	...	$(\cdot,\cdot,8)$	$(\cdot,\cdot,9)$
$(\cdot,\cdot,0)$	P_{insp}	0	0	...	0	0
$(\cdot,\cdot,1)$	$P_{insp} \cdot P_{not}$	0	0	...	0	0
$(\cdot,\cdot,2)$	$P_{insp} \cdot P_{not}^2$	0	0	...	0	0
...	...	...	...	...	...	...
$(\cdot,\cdot,8)$	$P_{insp} \cdot P_{not}^8$	0	0	...	0	0
$(\cdot,\cdot,9)$	$P_{insp} \cdot P_{not}^9$	0	0	...	0	0

(a) P Matrix for the action 'inspect',

State	$(\cdot,\cdot,0)$	$(\cdot,\cdot,1)$	$(\cdot,\cdot,2)$	...	$(\cdot,\cdot,8)$	$(\cdot,\cdot,9)$
$(\cdot,\cdot,0)$	0	P_{not}	0	...	0	0
$(\cdot,\cdot,1)$	0	0	P_{not}^2	...	0	0
$(\cdot,\cdot,2)$	0	0	0	...	0	0
...	...	...	...	...	...	...
$(\cdot,\cdot,8)$	0	0	0	...	0	P_{not}^9
$(\cdot,\cdot,9)$	0	0	0	...	0	P_{bound}

(b) P Matrix for the action 'do not inspect'.

Table 5: Augmented transition matrices for each action.

6.2. Optimal Policy

The solution to equation 5 reveals that a threshold-type policy is optimal. In fact, the results inform the decision-maker how many days should elapse before inspecting in case the current state has an optimal action of 'do not inspect'. We present two computational examples; the first assumes costs taken from the literature, and the latter uses a higher inspection cost to exemplify how a more diverse policy may arise.

The first experiment is performed for a segment assuming that the cost of a derailment is \$1.5M, and the cost of an inspection is \$1500 — assuming a 10km inspection at \$150/km (Soleimanmeigouni et al., 2016; Transportation Economics and Management Systems Inc, 2018). Repairs are assumed to cost \$1500 for red defects and \$1300 for yellow defects (He et al., 2015). Analyzing the data, we have observed that both red defects and some yellow defects require immediate repair, while 80% of yellow defects that are categorized as superficial may not. Hence, we assume that only more severe yellow defects, as well as all red defects will be repaired immediately.

In the first setting (with the inspection cost of \$150/km), the optimal MDP solution is to inspect the segment right away, no matter in which state the segment currently resides. This suggests our data contains segments that are prone to defects. However, we know that it may not always be feasible for companies to inspect every segment daily. In the case of limitations as such, we provide another approach in the next section.

For the second experiment, we increase the inspection cost to \$5000 (\$500/km) so that we can visualize a more diverse optimal policy. In this scenario, the threshold type policy

prescribes an inspection rule in such a way that it is optimal to inspect every time the segment has red defects under high load; to wait 1 day when the system has *i*) red defects and is under low load, or *ii*) yellow defects, but no red defects and is under high load; to wait 4 days when the system has yellow, but no red defects and is under low load; and to wait 5 days whenever there are not defects regardless of the load (see Figure 10). It should be noted that each country or railway company may have different regulations on the inspection frequency. In the case of regulation violations, the scheduling problem becomes a constrained problem that could be solved using the Linear Programming (LP) formulation of discounted MDPs given in Kallenberg (1989).
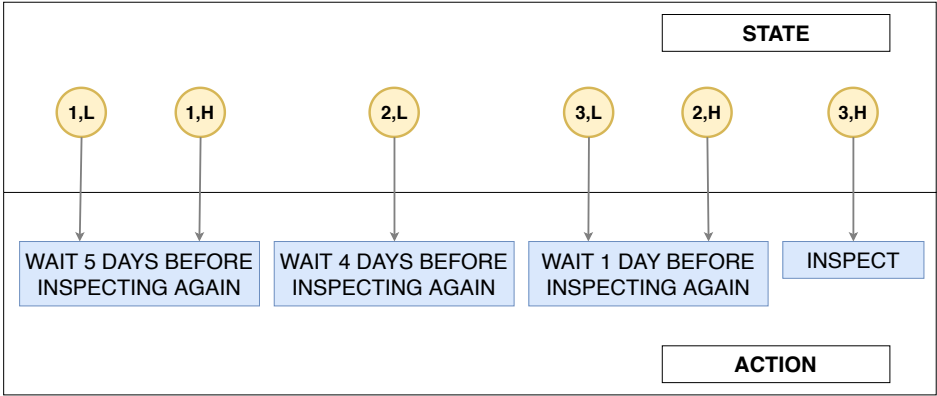

Figure 10: Inspection Scheduling Policy Representation from costs described in section 6.2 and $\alpha = 0.95$. Yellow nodes represent the input (current state), and blue nodes represent the output (actions to take). Note a threshold type policy is optimal.

The results are compared with a benchmark policy to measure the efficiency of the policy obtained by MDPs and are demonstrated in Figures 11a and 11c. We construct our benchmark policy based on the inspection schedule guidelines provided with the dataset. Inspections are performed twice a week on major segments and once a week on the other segments. Therefore, for the benchmark policy, we define a 7-day wait for states with low load (1, 2 and 3), and a 4-day wait for states with high load (4, 5, and 6). Note that neither the number nor the type of defects found previously have an impact on the policy for the benchmark policy.

Expected gains are on the magnitude of tens of thousands of dollars per year for each track segment, thus, showcasing the importance of reviewing how scheduling is currently

being assessed. The benchmark policy is approximately 100% and 23% more expensive than the policy resulting from the MDP approach for the respective cases in which inspection cost is set to \$150/km and \$500/km.

(a) Inspection cost=\$150/km, $\alpha = 0.95$

(b) Inspection cost=\$150/km, $\alpha = 0.5$

(c) Inspection cost=\$500/km, $\alpha = 0.95$
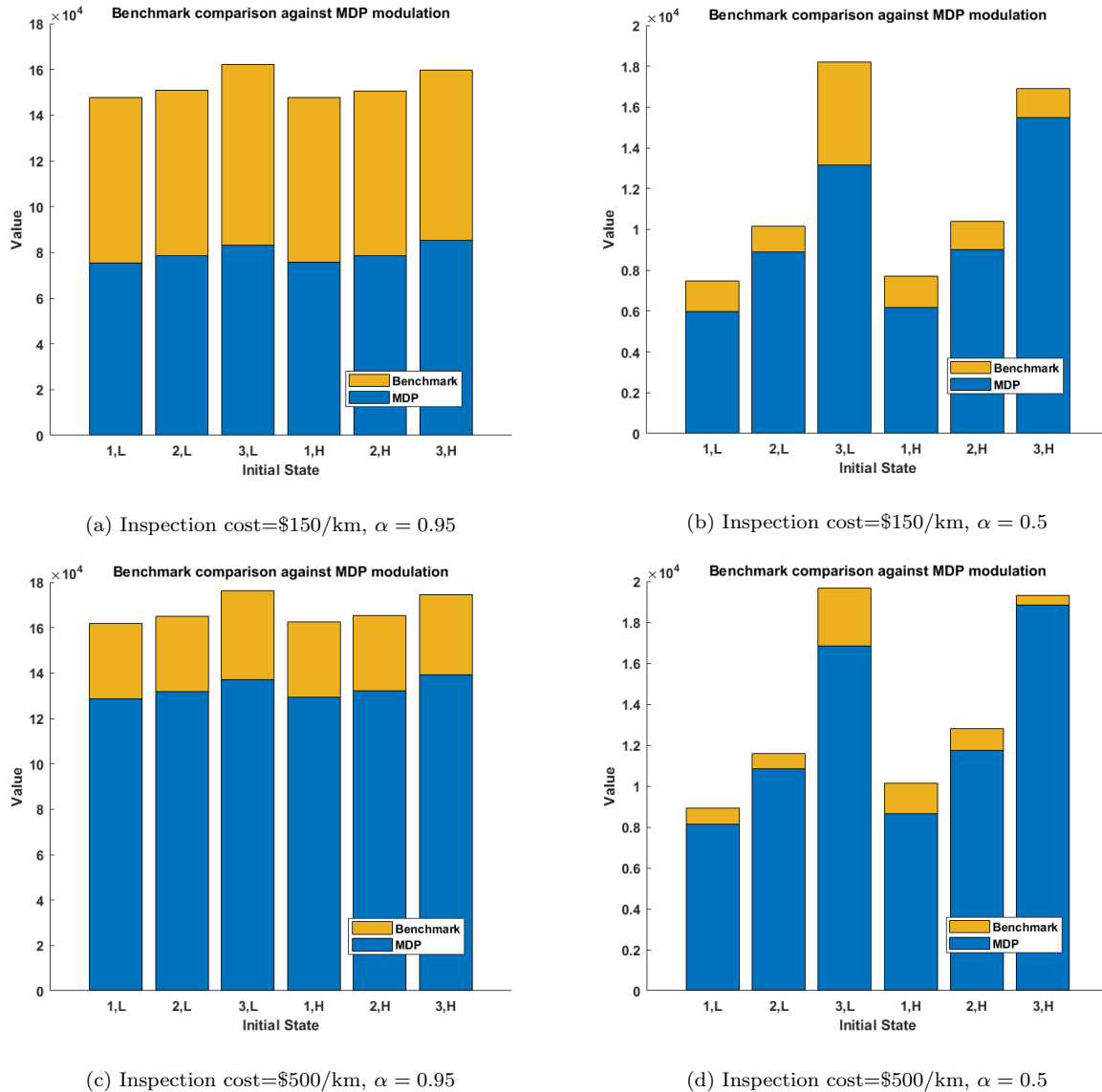
(d) Inspection cost=\$500/km, $\alpha = 0.5$

Figure 11: Bar plot representing the value using the benchmark policy and the optimal value obtained from the MDP formulation.

Figures 11b and 11d depict the results when the discount factor is set to $\alpha = 0.5$. It represents the short term savings by shifting to the MDP policy from the benchmark policy. As expected, the values are a lot more dependent on the initial state, as the discount factor has a higher impact on the convergence. The benchmark costs are approximately between

12% and 35% more expensive than the MDP policy when the inspection cost is \$150/km and between 2% and 18% when the inspection cost is \$500/km .

7. Segments as Restless Bandits

The results obtained in Section 6 provide decision-makers with a detailed policy they should follow when deciding whether to inspect or not to inspect a segment. However, crew limitations may cause the policy to be inadmissible. In this case, the problem can be formulated as the following constrained mathematical program

$$\lim_{N \to \infty} \min \left\{ E \left[\sum_{t=0}^{N} \alpha^t \sum_{i=1}^{n} c_i(s_i(t), a_i(t)) \right] \right\}$$

$$\text{s.t.} \sum_{i=1}^{n} a_i(t) \leq m, t \in \{1, 2, ..., N\}$$

$$a_i \in \{0, 1\} \forall i,$$

where $s_i(t)$ denotes the state of the ith segment at time t, $a_i(t)$ denotes the binary variable representing if segment i is inspected or not at time t, and $c_i(\cdot, \cdot)$ denotes the expected cost incurred at segment i when the state and action information are given.

Bandit problems are mathematical models to optimally allocate limited efforts in various competing projects so that maximum reward or minimum cost is achieved under uncertainty (Gittins et al., 2011). Original bandit problems assume no change of state and zero rewards or costs for a particular segmente when it is not inspected. However, in the inspection scheduling problem, this is not the case. A variation of the original bandit problem, called the restless bandit problem, allows for such evolution of state and passively received reward or cost (Gittins et al., 2011). Furthermore, this framework continuously updates the transition matrices, maintaining and improving the system robustness.

Assume there are n segments to be inspected, and each segment has an initial augmented state $s_i \in \mathcal{S}$, with $|\mathcal{S}| = 60$. Each state represents the level of deterioration of the rail segment at the beginning of the period and the time passed since the last inspection, as described in Section 6.1. Let us further assume that there are m crews available to inspect n segments. Other parameters include **a**, a vector of actions for all states, and $\alpha \in [0, 1]$, the discount

factor. Finally, let us assume that the initial probability transition matrices for the actions 'inspect' and 'do not inspect' are the ones obtained in section 6.1. If m, the number of crews available, is larger than n, all segments can be inspected and the problem becomes unconstrained. On the other hand, if $m < n$, this problem can be characterized as a restless bandit problem.

The restless bandit formulation chooses the best n of m competing segments optimally by using a priority index called Whittle index if the problem is indexable. We will develop the methodology below.

Since the constraint on the crew number can be bounded as

$$\sum_{t=0}^{\infty} \alpha^t \sum_{i=1}^{n} a_i(t) \leq \frac{m}{1-\alpha}, t \in \{1, 2, ..., N\},$$

by relaxing the activation constraint, we can write the Lagrangian dual function as

$$\mathcal{L}(\lambda) = \min_{a_i} \left\{ E\left[\sum_{t=0}^{\infty} \alpha^t \sum_{i=1}^{n} c_i(s_i(t), a_i(t)) + \lambda \cdot \left(\sum_{t=0}^{\infty} \alpha^t \sum_{i=1}^{n} a_i(t) - \frac{m}{1-\alpha} \right) \right] \right\}$$

$$= \min_{a_i} \left\{ E\left[\sum_{t=0}^{\infty} \alpha^t \sum_{i=1}^{n} \left(c_i(s_i(t), a_i(t)) + \lambda a_i(t) \right) \right] \right\} - \lambda\left(\frac{m}{1-\alpha} \right)$$

$$= \min_{a_i} \left\{ \sum_{t=0}^{\infty} \alpha^t \cdot E\left[\sum_{i=1}^{n} \left(c_i(s_i(t), a_i(t)]) + \lambda a_i(t) \right) \right] \right\} - \lambda\left(\frac{m}{1-\alpha} \right)$$

s.t. $a_i \in \{0, 1\} \forall i$.

This problem can be decoupled for each segment, after disregarding the last term, which is a constant.

$$\mathcal{C}_i(\lambda) = \min \left\{ E\left[\sum_{t=0}^{\infty} \alpha^t (c_i(s_i(t), a_i(t)) + \lambda a_i(t)) \right] \right\}$$

s.t. $a_i \in \{0, 1\} \forall i$.

The set of all states for which it is optimal to choose $a = 0$ when λ is fixed increases monotonically as λ increases. When $\lambda = 0$, all actions should be 'inspect' (based on the result from Section 6). As λ increases, there is a point at which it is better not to inspect.

Therefore, this problem is indexable (Gittins et al., 2011). The Whittle index is defined as

$$W_i(s_i) =$$

$$\inf \left\{ \lambda : E\left[c_i(s_i, 0) + \sum_{t=1}^{\infty} \alpha^t(c_i(s_i(t), a_i(t)) + \lambda \cdot a_i(t))\right] < \right.$$

$$\left. E\left[c_i(s_i, 1) + \lambda + \sum_{t=1}^{\infty} \alpha^t(c_i(s_i(t), a_i(t)) + \lambda \cdot a_i(t))\right]\right\}. \tag{7}$$

Inspection decisions involve a risk-based approach that incorporates the trade-off between the risk cost and the repair cost. The Whittle index, a well-established measure for restless bandits, ranks the importance of this trade-off.

To compute the Whittle indices from equation 7, we solve several MDP for iteratively increasing λ. Because the cardinality of both the state and action spaces is finite, value-iteration or policy-iteration algorithms can be used to find the optimal cost function for each MDP efficiently (Bertsekas, 2007). After calculating an index for each segment in the candidate segments as identified in Section 6, and sorting them from the highest to lowest, the decision maker should pick the first m segments with the highest indices for inspection. Finally, the transition kernels are updated for these m segments using the maximum likelihood estimation.

7.1. Example

We provide a simplified example in which 5 segments from the same cluster are being considered for inspection, but only two crews are available. Of course, this methodology also applies to more general cases in which segments come from different clusters.

The cost parameters are assumed to be the same as the ones given in Section 6.2, but with minor perturbations to characterize the particularities of each segment. In this scenario, inspections on segments B, and C cover 10 km of their length, while inspections on segments A, D, and E cover 7 km of their length at the rate of $150 per km inspected.

Segments have been inspected t days ago and their last observed state is described in Table 6a. Suppose that the initial belief for the transition matrices of this cluster is that they follow the ones in Tables 4a and 4b.

Segment	A	B	C	D	E
Derailment ($)	2M	2M	2M	1.5M	1.5M
Inspection ($)	1050	1500	1500	1050	1050
Red Repair ($)	1500	1500	1500	1400	1400
Yel. Repair ($)	1300	1200	1300	1300	1200
Last State	(3,L)	(3,H)	(2,L)	(2,H)	(1,L)
t	1	0	4	1	6

Segment	Augmented State	Whittle Index
A	(3,L,1)	9600
B	(3,H,0)	10100
C	(2,L,4)	2100
D	(2,H,1)	2500
E	(1,L,6)	10200

(a) Costs and state information for each segment.   (b) Whittle Indices for each segment.

Table 6: Example cost and state parameters for each of the five segments and the corresponding results.

7.1.1. Inspection policy

The calculated Whittle indices are presented in Table 6b. Indices are calculated in increments of 100. Providing such a table allows the decision-maker to quickly assess which segments should be prioritized in a situation where only a limited number of crews is available. The decision-maker should choose the segments with highest indices.

Table 6b suggests that inspecting segments B and E is the best decision in this scenario. The new states for segments B and E are observed, and t is set to 0. States for segments A, C, and D are updated to (3,L,2), (2,L,5), and (2,H,2), respectively.

7.1.2. Updating transition matrices

Lastly, the transitions matrices for segments B and E need to be updated, after the inspection is finished and the new state is known. Suppose the states after the inspection of segments B, and E are 5 and 2, respectively. Since the new t is 0, given that they have just been inspected, the corresponding augmented states are 5 (2,H,0), and 2 (2,L,0).

With two new inspections, we can update the frequencies of the augmented transition matrices. We normalize the matrices by the total number of inspections, including the last two inspections. Finally, we compute the updated probabilities for the transitions matrices following the procedure in Section 6.1, but directly using the augmented matrices instead (using MLE on the updated frequencies). These new transition matrices are now the updated ones for the corresponding cluster. Due to the strong law of large numbers, it is expected that they will tend to the "true" ones as more and more data are accumulated.

8. Conclusion

This study proposes robust approaches for defect prediction in railways when faulty data are present or detailed data are not available. Moreover, it proposes an integration between such predictions and the problem of optimally scheduling inspections and maintenances. The accuracy of the prediction models indicates that detailed data may not be necessary if simple yet accurate data are available. Additionally, we present adaptations for higher levels of risk-aversion. Furthermore, a threshold-type policy is shown to be optimal for scheduling when each segment has a dedicated inspector, i.e., there are no crew limitations. A state-augmentation approach is used to account for the partial observability of the system. Results indicate the potential for considerable savings by using the optimal policy. Lastly, under an additional constraint of crew limitations, we show that the problem is indexable. We provide the index equation and describe the optimal policy, as well as how to update the transition matrices after each period.

In this study, we do not account for the layout of the segments and how the geometry of the system can play a role in the assignment of crews for inspection and maintenance. Moreover, limitations such as available equipment, ongoing rail traffic, inspection and repair durations, possible routes, etc. also form other constraints affecting detailed scheduling. This paper presents a solution to the long-term master policy and schedule structuring problem where crews are allowed to inspect any given segment within a region, independent of where they are currently located. Future research will integrate long-term master policy and schedules to short-term detailed schedule planning.

9. ACKNOWLEDGEMENTS

References

Anderson, R. T., Barkan, C. P., 2005. Derailment probability analysis and modeling of mainline freight trains. In: Proceedings of 8th International Heavy Haul Railway Conference. pp. 88–98.

Anderson, T. W., Goodman, L. A., 1957. Statistical inference about Markov chains. The Annals of Mathematical Statistics, 89–110.

Andrade, A. R., Teixeira, P. F., 2016. Exploring different alert limit strategies in the maintenance of railway track geometry. Journal of Transportation Engineering 142 (9), 04016037/1–9.

Azizyan, M., Singh, A., Wasserman, L., 2014. Feature Selection For High-Dimensional Clustering. arXiv e-prints (1406.2240), 1–11.

Bartlett, M. S., 1951. The frequency goodness of fit test for probability chains. In: Mathematical Proceedings of the Cambridge Philosophical Society. Vol. 47. Cambridge University Press, pp. 86–95.

Bertsekas, D. P., 2005. Dynamic programming and optimal control. Vol. 1. Athena Scientific Belmont, Massachusetts.

Bertsekas, D. P., 2007. Dynamic programming and optimal control. Vol. 2. Athena Scientific Belmont, Massachusetts.

Bioch, J. C., Popova, V., 2002. Monotone decision trees and noisy data. ERIM Report Series, 1–23.

Breiman, L., Friedman, J. H., Olshen, R. A., Stone, C. J., 1984. Classification and Regression Trees. Wadsworth.

Budai-Balke, G., Dekker, R., Kaymak, U., 2009. Genetic and memetic algorithms for scheduling railway maintenance activities. Econometric Institute, Erasmus University Rotterdam (EI 2009-30), 1–23.

Burden, F., Winkler, D., 2008. Bayesian regularization of neural networks. In: Artificial Neural Networks. Vol. 458. Springer, pp. 23–42.

Camci, F., 2014. The travelling maintainer problem: integration of condition-based maintenance with the traveling salesman problem. Journal of the Operational Research Society 65 (9), 1423–1436.

Cárdenas-Gallo, I., Sarmiento, C. A., Morales, G. A., Bolivar, M. A., Akhavan-Tabatabaei, R., 2017. An ensemble classifier to predict track geometry degradation. Reliability Engineering & System Safety 161, 53–60.

Chen, S.-K., Ho, T.-K., Mao, B.-H., Bai, Y., 2014. A bi-objective maintenance scheduling for power feeding substations in electrified railways. Transportation Research Part C: Emerging Technologies 44, 350 – 362.

Clark, R., 2004. Rail flaw detection: overview and needs for future developments. NDT & E International 37 (2), 111 – 118.

Consilvio, A., Febbraio, A. D., Sacco, N., 2016. Stochastic scheduling approach for predictive risk-based railway maintenance. In: 2016 IEEE International Conference on Intelligent Rail Transportation (ICIRT). pp. 197–203.

D'Ariano, A., Meng, L., Centulio, G., Corman, F., 2019. Integrated stochastic optimization approaches for tactical scheduling of trains and railway infrastructure maintenance. Computers & Industrial Engineering 127, 1315 – 1335.

Doric, D., El Cadi, A., Hanafi, S., Mladenovic, M., Artiba, A., 2017. Clustering approach in maintenance of capillary railway network. In: 4th International Conference on Variable Neighborhood Search. Vol. 58. pp. 239 – 246.

Eberhart, R., Kennedy, J., 1995. A new optimizer using particle swarm theory. In: Proceedings of the Sixth International Symposium on Micro Machine and Human Science, MHS'95. IEEE, pp. 39–43.

Fan, P., Seungmo, K., Xiaopeng, L., Yanfeng, O., Kamalesh, S., Dharma, A., 2011. A heuristic approach to the railroad track maintenance scheduling problem. Computer-Aided Civil and Infrastructure Engineering 26 (2), 129–145.

Friedman, J., Hastie, T., Tibshirani, R., 2001. The elements of statistical learning. Vol. 1. Springer series in statistics New York, NY, USA:.

Ghofrani, F., He, Q., Goverde, R. M., Liu, X., 2018. Recent applications of big data analytics in railway transportation systems: A survey. Transportation Research Part C: Emerging Technologies 90, 226 – 246.

Gittins, J., Glazebrook, K., Weber, R., 2011. Multi-armed bandit allocation indices. John Wiley & Sons.

Güler, H., 2014. Prediction of railway track geometry deterioration using artificial neural networks: a case study for Turkish state railways. Structure and Infrastructure Engineering 10 (5), 614–626.

Hamid, A., Gross, A., 1981. Track-quality indices and track degradation models for maintenance-of-way planning. Transportation Research Board 802, 2–8.

He, Q., Li, H., Bhattacharjya, D., Parikh, D. P., Hampapur, A., 2015. Track geometry defect rectification based on track deterioration modelling and derailment risk assessment. Journal of the Operational Research Society 66 (3), 392–404.

Heinicke, F., Simroth, A., Scheithauer, G., Fischer, A., 2015. A railway maintenance scheduling problem with customer costs. EURO Journal on Transportation and Logistics 4 (1), 113–137.

Hiscott, R., 1981. Chi-square tests for Markov chain analysis. Journal of the International Association for Mathematical Geology 13 (1), 69,80.

Hubert, L. J., Levin, J. R., 1976. A general statistical framework for assessing categorical clustering in free recall. Psychological Bulletin 83 (6), 1072–1080.

Jiang, Y., Zhang, Z., Voorhis, T. V., McCalley, J. D., 2003. Risk-based maintenance optimization for transmission equipment. In: In Proc. of 12th Annual Substations Equipment Diagnostics Conference. pp. 1191–1200.

Kallenberg, L., 1989. Markov decision processes. European Journal of Operational Research 39 (1), 1 – 16.

Khalouli, S., Benmansour, R., Hanafi, S., 2016. An ant colony algorithm based on opportunities for scheduling the preventive railway maintenance. In: 2016 International Conference on Control, Decision and Information Technologies (CoDIT). pp. 594–599.

Khan, S. S., Ahmad, A., 2004. Cluster center initialization algorithm for k-means clustering. Pattern Recognition Letters 25 (11), 1293 – 1302.

Lannez, S., Artigues, C., Damay, J., Gendreau, M., 2015. A railroad maintenance problem solved with a cut and column generation metaheuristic. Networks 66 (1), 40–56.

Lidén, T., Joborn, M., 2017. An optimization model for integrated planning of railway traffic and network maintenance. Transportation Research Part C: Emerging Technologies 74, 327 – 347.

Liu, X., Saat, M. R., Barkan, C. P., 2012. Analysis of causes of major train derailment and their effect on accident rates. Transportation Research Record 2289 (1), 154–163.

Luan, X., Miao, J., Meng, L., Corman, F., Lodewijks, G., 2017. Integrated optimization on train scheduling and preventive maintenance time slots planning. Transportation Research Part C: Emerging Technologies 80, 329 – 359.

Martey, E. N., Ahmed, L., Attoh-Okine, N., 2017. Track geometry big data analysis: A machine learning approach. In: Big Data (Big Data), 2017 IEEE International Conference on. IEEE, pp. 3800–3809.

Merrick, J. R., Soyer, R., 2015. Semiparametric bayesian optimal replacement policies: application to railroad tracks. Applied Stochastic Models in Business and Industry 33 (5), 445–460.

Moridpour, S., Mazloumi, E., Hesami, R., 2017. Application of artificial neural networks in predicting the degradation of tram tracks using maintenance data. In: Applied Big Data Analytics in Operations Management. IGI Global, pp. 30–54.

Nishi, T., Muroi, Y., Inuiguchi, M., 2011. Column generation with dual inequalities for railway crew scheduling problems. Public Transport 3 (1), 25–42.

Peralta, D., Bergmeir, C., Krone, M., Galende, M., Menéndez, M., Sainz-Palmero, G. I., Bertrand, C. M., Klawonn, F., Benítez, J. M., 2018. Multiobjective optimization for railway maintenance plans. Journal of Computing in Civil Engineering 32 (3), 04018014/1–11.

Pour, S. M., Benlic, U., 2017. Clustering of maintenance tasks for the Danish railway system. In: Madureira, A. M., Abraham, A., Gamboa, D., Novais, P. (Eds.), Intelligent Systems Design and Applications. Springer International Publishing, Cham, pp. 791–799.

Pour, S. M., Drake, J. H., Ejlertsen, L. S., Rasmussen, K. M., Burke, E. K., 2018. A hybrid constraint programming/mixed integer programming framework for the preventive signaling maintenance crew scheduling problem. European Journal of Operational Research 269 (1), 341 – 352.

Puterman, M. L., 1994. Markov Decision Processes: Discrete Stochastic Dynamic Programming, 1st Edition. John Wiley & Sons, Inc., New York, NY, USA.

Ross, S., 1992. Applied Probability Models with Optimization Applications. Dover Books on Advanced Mathematics. Dover Publications.

Sharma, S., Cui, Y., He, Q., Mohammadi, R., Li, Z., 2018. Data-driven optimization of railway maintenance for track geometry. Transportation Research Part C: Emerging Technologies 90, 34 – 58.

Simson, S., Ferreira, L., Murray, M., 2000. Rail track maintenance planning: An assessment model. Transportation Research Record: Journal of the Transportation Research Board 1713, 29–35.

Soleimanmeigouni, I., Ahmadi, A., Letot, C., Nissen, A., Kumar, U., 2016. Cost-based optimization of track geometry inspection. In: Proceedings of the 11th World Congress on Railway Research. pp. 1–7.

Thomas, J. L., Jones, G. N., Scarinci, I. C., Mehan, D. J., Brantley, P. J., 2001. The utility of the CES-D as a depression screening measure among low-income women attending primary care clinics. The International Journal of Psychiatry in Medicine 31 (1), 25–40.

Transportation Economics, Management Systems Inc, 2018. Northern Michigan Rail Ridership Feasibility and Cost Estimate Study. Tech. rep., The Groundwork Center For Resilient Communities, Grant Fiduciary: Bay Area Transportation Authority.

Turner, C., Tiwari, A., Starr, A., Blacktop, K., 2016. A review of key planning and scheduling in the rail industry in Europe and UK. Proceedings of the Institution of Mechanical Engineers, Part F: Journal of Rail and Rapid Transit 230 (3), 984–998.

Wang, L., An, M., Qin, Y., Jia, L., 2018. A risk-based maintenance decision-making approach for railway asset management. International Journal of Software Engineering and Knowledge Engineering 28 (04), 453–483.