Condition-based Infrastructure Maintenance Optimization with Combinatorial Q-Learning

Akira Tanimoto[1,2,3,a)]

Abstract: Planning of infrastructure maintenance is a large-scale optimization problem of planning when and which components to perform maintenance so as to keep the whole infrastructure in good condition with a minimal maintenance cost. Recent advances of condition monitoring techniques enabled timely maintenance in response to the condition of each part regardless of its age. In addition to the condition, the spatial structure is also important for efficiency in infrastructure maintenance since the traveling cost and/or setup cost can be saved by simultaneous maintenance of neighboring components, which is called economic dependency. This optimization problem naively has the high computational complexity of $O(2^{nH})$, where n is the number of components and H is the planning horizon, and also the predictive modeling of degradation is another big issue. To solve this problem efficiently at scale, our proposed method utilizes two kinds of dynamic programming for temporal and spatial scalability and consequently enjoys $O(n)$ complexity at each time step. For temporal scalability, we employ a direct modeling approach for the action value of maintenance instead of modeling degradation, namely, Q-learning. And for spatial scalability, we exploit locality in economic dependency via a reasonable approximation of the Q-function. A considerable baseline approach is that one divides the whole infrastructure into fixed groups of neighboring components beforehand and decides to perform maintenance or not for all the components in each group at each time step. On the other hand, our scalable method enables fully combinatorial optimization for each component at each time step. We show the advantage of our method in a simulated environment, and the resulting maintenance history intuitively illustrates the benefit of our dynamic grouping approach. We also show that our method has a kind of interpretability in the optimization at each time step.

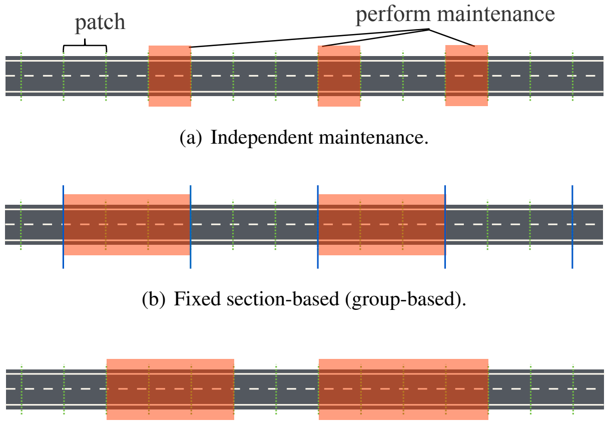
Keywords: Infrastructure maintenance planning, Reinforcement learning, Q-learning, Combinatorial optimization, Dynamic programming

1. Introduction

We consider an infrastructure maintenance planning problem for road surfaces of highways, water, oil, and gas pipelines, and so on. In each discretized time step, the maintenance decision maker considers which components, or the small patches of the road surface, should be maintained based on regularly observed condition of each component. If some patches are almost deteriorated and geospatially neighboring, simultaneous maintenance is economical as shown in Fig. 1.

A huge maintenance cost is paid to keep the infrastructure in good condition since the condition of an infrastructure is critical in terms of safety, conformity, and prevention of economic loss caused by emergent corrective maintenance or availability loss. We focus to reduce the total cost, i.e., the sum of the maintenance cost and the condition cost (risk) caused by a deteriorated infrastructure.

Total cost minimization by maintenance planning is well investigated in literature [7]. For multi-component systems, those that have multiple maintenance targets, so-called economic de-



(a) Independent maintenance.

(b) Fixed section-based (group-based).

(c) Dynamic grouping by combinatorial action optimization (proposed).

Fig. 1 Comparison of road maintenance policies. Performing maintenance of longer sections that cover multiple deteriorated patches may cost less in the long run. That is, when multiple components are maintained simultaneously, overall maintenance costs are reduced since traveling cost of the maintenance team, and/or setup costs are saved; this is called economic dependency. Thus, fixed section-based maintenance policy (b) is preferable than independent maintenance policy (a). Proposed dynamic grouping policy (c) is computationally expensive but is more flexible than the two baselines. Results of our experiment shows the advantage of our approach and the resulting maintenance history illustrates why the proposed policy is notably better.

[1] NEC, 1753 Shimonumabe, Nakahara Ward, Kawasaki, Kanagawa 211-0011, Japan

[2] Kyoto University, Yoshida-honmachi, Sakyo-ku, Kyoto 606-8501, Japan

[3] Riken AIP, Nihonbashi 1-chome Mitsui Building, 15th floor, 1-4-1 Nihonbashi, Chuo-ku, Tokyo 103-0027, Japan

[a)] a.tanimoto@nec.com

pendency of the targets and group-based maintenance is often discussed [3], [12]. Infrastructure maintenance can also be regarded as multi-component maintenance by considering small patches as components. In road maintenance, for example, cost savings can be realized by maintaining larger sections instead of small patches [1], [12]. In [13], maintenance optimization technique for infrastructure network is proposed. They formalized a special type of economic dependency for infrastructure network, namely, the network topology dependency (NTD), and proposed an optimization method under the benefit of maintenance for each component given. NTD assumption reflects the locality of economic dependency in infrastructure maintenance; i.e., the cost reduction is realized only when the neighboring components are maintained simultaneously. To consider complex economic dependency such as NTD, combinatorial optimization is required, and the computational complexity is high. The proposed optimization method in [13] exploits the submodularity in NTD for computational efficiency.

These maintenance optimization methods for multi-component systems are mostly built on the time-based maintenance (TBM) setting, in which each component has a predefined lifetime. Thus, the benefit of maintenance for each component can be calculated but the uncertainty in the deterioration process is not considered. On the other hand, thanks to the recent development of health condition monitoring technologies, the actual condition of each component of an infrastructure is becoming observed almost timely. To name a few, image processing [2] and sensor networks [8] for road surfaces, and fiber optic sensing for pipelines [6], [9]. These sensing technologies contribute to cost savings since only deteriorated components are maintained regardless of their age, which policy is known as condition-based maintenance (CBM). Note that, CBM includes a wide range of maintenance concepts, which are characterized as predictive maintenance helped by condition monitoring technologies.

Optimization for multi-component CBM is not straightforward due to the economic dependency and the uncertainty in condition degradation. Actually, the optimization for that setting is computationally much harder than TBM when taking the uncertainty into account. Studies for condition-based maintenance of large-scale multi-component systems such as those for infrastructures are limited. Existing work in this context [11], [15], [16] for systems such as those for heavy vehicles assumes simple economic dependency, i.e., constant maintenance costs or cost reductions regardless of the number of components or which components are to be maintained. Since infrastructures are geospatially distributed systems with large numbers of components, the locality of economic dependency such as NTD should be considered.

A simple heuristic approach to avoid the whole combinatorial optimization in respect to locality is dividing the whole infrastructure into larger local sections in advance, which is called fixed section-based maintenance policy as illustrated in Fig. 1(b). Although, this simplified approach lacks flexibility in optimization, which leads to limited performance.

To fully consider the local economic dependency and optimize large-scale maintenance actions efficiently, we employ two kinds of dynamic programming techniques for temporal and spatial scalability. For temporal scalability, we employ a direct modeling approach of a cost-benefit evaluator, that is, Q-learning [17]. Q-learning aims to learn the total cost-benefit in the long run under the observed condition as the state-action value function (known as the Q-function), $Q(s, a)$. Once Q-function is learned, one can easily evaluate the maintenance action without assessing the uncertain future degradation. Moreover, for spatial scalability of combinatorial optimization of actions, we propose an approximated Q-function model and a linear-time optimization algorithm by exploiting the locality in the economic dependency. The scalable action optimization is necessary also for learning Q-function since the Q-learning requires the optimal value $\min_a Q(s, a)$ in each learning iteration. Although our Q-function is simple, realized dynamic grouping of neighboring maintenance targets in Fig. 1(c) turned out to be significantly better than the fixed section-based approach.

In addition to the performance, our proposed method has a kind of interpretability. Since maintenance decision makers often have the responsibility for safety, the interpretability of optimized solution matters. In our parametrized Q-function, the maintenance benefit for each component and cost are separated. And thus, the estimated benefit for each component can be shown in the same figure with the condition of each component, which enables the decision makers to assess the cost-benefit tradeoff. The detailed discussion is in Section 5.

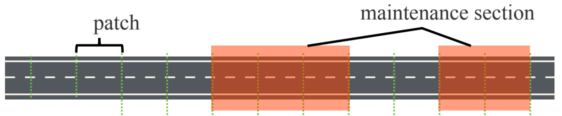
We compare our dynamic grouping approach and the fixed section-based approach in the experiment since the independent maintenance policy shown in Fig. 1(a) is included in the fixed section-based maintenance policy as the setting of section length (window width) is one. The optimized maintenance history gives an intuitive explanation of the advantage of determining groups dynamically.

For the geospatial structure of the maintenance targets, we focus on one-dimensional (1-D) cases such as highways and pipelines, which is the simplest case to demonstrate the advantage of our approach. In addition, most part of the highway, for example, is one-dimensional. For the highway network, a combined policy of fixed section-based for intersection and branching part and dynamic grouping for the rest would work.

2. Problem Formulation

We determine when and which maintenance targets (small patches of road or pipeline) should be maintained to minimize the cumulative cost including future maintenance cost and condition cost. We assume the current cost is given explicitly as the cost function $\mathrm{Cost}(\boldsymbol{s}, \boldsymbol{a})$, where $\boldsymbol{s} = \{s_i\}_i, s_i \in \mathbb{R}$ is the state (condition) and $\boldsymbol{a} = \{a_i\}_i, a_i \in \{0, 1\}$ is the action taken at each time step ($a_i = 1$ represents that the maintenance is performed for the i-th patch).

The final goal is as follows. At each time step t, given the observed states (or the condition) $\boldsymbol{s}_t \in \mathbb{R}^n$, where n is the number of maintenance targets, we determine which targets are to be maintained to minimize the expected total cost in the long run in regard to future actions assumed to be optimized. Thus, the optimal action for the time step t is



patch id	…	i	i+1	i+2	i+3	i+4	…			…
$s_{t,i'}$	…	$s_{t,i}$	$s_{t,i+1}$	$s_{t,i+2}$	$s_{t,i+3}$	$s_{t,i+4}$	…			…
$a_{t,i'}$	…	0	1	1	1	0	…	1	1	…
working cost	…	0	c_w	c_w	c_w	0	…	c_w	c_w	…
traveling cost	…	0	c_t			0	…	c_t		…
subtotal cost	…	0	$c_t + 3c_w$			0	…	$c_t + 2c_w$		…

Fig. 2 Cost of maintenance action assumed in one-dimensional targets environment.

$$\boldsymbol{a}_t^* = \mathop{\arg\min}_{\boldsymbol{a}\in\Gamma(t)\subseteq\{0,1\}^n} \Big\{ \mathrm{Cost}(\boldsymbol{s}_t, \boldsymbol{a}) + \min_{\{\boldsymbol{a}_{t'}\}_{t+1}^{t+H}} \sum_{t'=t+1}^{t+H} \beta^{t'-t} \mathop{\mathbb{E}}_{\boldsymbol{s}_{t'}|\boldsymbol{s}_t,\boldsymbol{a}_t,\ldots,\boldsymbol{a}_{t'-1}} [\mathrm{Cost}(\boldsymbol{s}_{t'}, \boldsymbol{a}_{t'})] \Big\}, \tag{1}$$

where $\beta \in [0, 1]$ is the discount parameter, $H \in \mathbb{N} \cup \{\infty\}$ is the prediction horizon, and $\Gamma(t)$ is the feasible set of actions. $a_{t,i}$ is the maintenance action for i-th target at t. In the following sections, we assume $\Gamma(t) = \{0, 1\}^n$.

For the cost function, it is reasonable to be separated into maintenance (action) cost and condition (state) cost; namely,

$$\mathrm{Cost}(\boldsymbol{s}, \boldsymbol{a}) = C_a(\boldsymbol{a}) + C_s(\boldsymbol{s}).$$

And the local economic dependency in action cost is formalized as follows.

$$C_a(\boldsymbol{a}) := a_1(c_w + c_t) + \sum_{i=2}^{n} a_i\{c_w + (1 - a_{i-1})c_t\}, \tag{2}$$

where c_t and c_w are given constants that represent the traveling costs occurring once for neighboring patches maintained simultaneously and working costs for each patch, respectively. Fig. 2 illustrates the calculation of the action cost. The interaction term $-a_i a_{i-1} c_t$ represents the local economic dependency, which comes from the savings of traveling costs. Although only the dependency of one-neighboring components is modeled in (2), one can easily extend the length of the locality considered, i.e., including the term $-a_i a_{i-k} c_k$, and the computational complexity would be $O(n2^k)$ in our optimization approach. We restrict $k = 1$ for simplicity.

For the state (condition) cost function, we assume the independence of each component. The dependent state cost setting is also studied as the stochastic dependency in [16], though we focus on economic dependency. For the state cost of each component, it is reasonable to assume some non-decreasing function. In our experiment, we set the following hinge cost.

$$C_s(\boldsymbol{s}) := c_s \sum_{i}^{n} (s_i - \alpha)_+, \tag{3}$$

where $(x)_+ := \max\{x, 0\}$, c_s and α are given constants.

In addition, we assume plenty of training data $\mathcal{D}$ of maintenance history under some unknown policy given instead of accurate prediction of the condition degradation or the benefit of maintenance for each component.

3. Related Work

Condition-based infrastructure maintenance planning at scale is not investigated well. We introduce some related work and make the difference from our setting clear.

3.1 Multi-Component Maintenance Optimization

The most related area is multi-component maintenance optimization. In [12], various types of dependency of components including economic dependency are reviewed. Especially, network topology dependency in [13] is the most related setting to our local economic dependency. However, in this area, basically time-based maintenance is assumed, in which the maintenance benefit is given or easily calculated since the aging process is deterministic. To the best of our knowledge, condition-based multi-component maintenance at scale is a novel setting.

3.2 Model-based Combinatorial Optimization

While we adopted model-free approach, Q-learning, model-based approach is also considerable. In model-based approach, first, the transition model $\boldsymbol{s}_t = M(\boldsymbol{s}_{t-1}, \boldsymbol{a})$ is estimated, and then, based on the estimated model, the action optimization and the future prediction to some prediction horizon are iteratively performed. Since this approach is computationally complexed, existing work [11], [15], [16] assumes simple economic dependencies. In other areas out of maintenance, rebalancing in bike sharing is thought to be a maintenance task in that bike inventory in each 2-D distributed station is maintained to be not empty or full. In [10], combinatorial optimization based on predicted values for such a problem; however, stations are clustered in advance. The advantage of our approach is determining maintenance groups dynamically, i.e., combinatorial optimization is performed in every time step.

4. Method

The general framework we adopted for this problem is fitted-Q learning [14] described in Algorithm 1. The difference from the original work is the combinatorial optimization in the loop $\min_{\boldsymbol{a}'} Q(\boldsymbol{s}, \boldsymbol{a}')$ and the model of the Q-function tailored for our problem setting.

Fitted Q-learning is an off-policy Q-learning method, namely, the training data generated from unknown policy is only needed for training, while on-policy learning updates its parameters with performing experiment in the real environment. In mission critical systems such as infrastructure maintenance, online update is not feasible, and the maintenance history by human experts is often available. In addition, the future value $\min_{a'} Q_\theta(s_{t+1}, a')$ is not differentiable with respect to θ due to the discrete optimization in a. Thus, in fitted-Q learning, the derivative is taken only for the current value and the future value is fixed.

4.1 Q-function Approximation for Maintenance Optimization

The key enabler of scalable optimization in this paper is to consider only local interactions of actions. Considering that Q-

Algorithm 1 Fitted-Q for maintenance optimization

```
Input: $\mathcal{D} = \{s_t, a_t, r_t, s_{t+1}\}_t, \beta, \mathrm{Cost}(\cdot,\cdot)$.
  initialize $\theta$.
  $k \leftarrow 0$
  while no convergence is met do
    get $(\boldsymbol{s}_t, \boldsymbol{a}_t, \boldsymbol{s}_{t+1})$ from $\mathcal{D}$ in random order.
    $y \leftarrow \mathrm{Cost}(\boldsymbol{s}_t, \boldsymbol{a}_t) + \beta \min_{\boldsymbol{a}'} Q_\theta(\boldsymbol{s}_{t+1}, \boldsymbol{a}')$
    $L_{\theta'} := \frac{1}{2}(y - Q_{\theta'}(\boldsymbol{s}_t, \boldsymbol{a}_t))^2$
    $\gamma_k = (2 + k)^{-1/2}$
    $\theta \leftarrow \theta - \gamma_k \frac{\mathrm{d}L_{\theta'}}{\mathrm{d}\theta'}|_{\theta'=\theta}$
    $k \leftarrow k + 1$
  end while
  return $\theta$
```

function is the approximation of (1) except for current state cost $C_s(\boldsymbol{s}_t)$, Q-function consists of current action cost $C_a(\boldsymbol{a})$ and discounted future costs, which is induced by degraded components. From this observation, we separate Q-function into action cost and the condition evaluation. We assume complete maintenance, i.e., the evaluated state values of maintained components are assumed to be constant, which we set to zero. And also, we approximate the state evaluation independent samely as in [13]. That is, the benefit of performing maintenance is assigned to each component separately. The resulting formulation is the following.

$$Q_{\boldsymbol{\theta}}(\boldsymbol{s}, \boldsymbol{a}) := C_a(\boldsymbol{a}) + \sum_{i}^{n} (1 - a_i) q(s_i; \boldsymbol{\theta}) + \theta_0. \qquad (4)$$

This separation of state evaluation is actually an approximation since neighboring degraded components in the future are easier to perform maintenance than distributed ones, which is not represented in our separate formulation. However, we adopted separated state evaluation mainly for both the simplicity of q function and the interpretability. In this formulation, the value $q(s_i)$ can be interpreted as the priority of performing maintenance for i-th component. The detailed discussion is in Section 6.1.

4.2 Q-function Optimization via Dynamic Programming

Our approximated Q-function can be optimized with respect to the action in linear time via dynamic programming. This is because, thanks to the locality of economic dependency, the optimal action of a patch depends only on the optimal action of the neighbors; i.e., it has the substructure optimality as shown below.

First, we define the partial value function $v_i(a)$ as

$$\begin{aligned} v_1(a) :=& a_1(c_w + c_t) + (1 - a_1) q(s_1), \\ v_i(a_i) :=& \min_{a_1,\ldots,a_{i-1}} \Big\{ a_i(c_w + c_t) + (1 - a_i) q(s_1) \\ &+ \sum_{i'=2}^{i} a_{i'} \{c_w + (1 - a_{i'-1}) c_t\} \\ &+ \sum_{i'+1}^{i} (1 - a_{i'}) q(s_{i'}) \Big\}. \end{aligned}$$

Note that, the minimization of $v_n(a)$ is equivalent to that of the whole Q-function.

$$\min_{a_n} v_n(a_n) + \theta_0 = \min_{a_1,\ldots,a_n} Q_\theta(\boldsymbol{a}, \boldsymbol{s}).$$

Algorithm 2 Dynamic programming for optimizing $\boldsymbol{a}$

```
Input: $\boldsymbol{s}_t, \boldsymbol{\theta}$.
Output: $\boldsymbol{a}^* = \arg\min_{\boldsymbol{a}' \in \{0,1\}^n} Q_\theta(\boldsymbol{s}_t, \boldsymbol{a}')$
  % forward step
  $v_1(a_1 = 0) = q(s_1; \boldsymbol{\theta})$
  $v_1(a_1 = 1) = c_w + c_t$
  for $i = 2, ..., n$ do
    $a_{i-1}(a_i = 0) = \arg\min_{a' \in \{0,1\}} v_{i-1}(a_{i-1} = a') + q(s_i; \boldsymbol{\theta})$
    $v_i(a_i = 0) = \min_{a' \in \{0,1\}} v_{i-1}(a_{i-1} = a') + q(s_i; \boldsymbol{\theta})$
    $a_{i-1}(a_i = 1) = \arg\min_{a' \in \{0,1\}} v_{i-1}(a_{i-1} = a') + (1 - a')c_t + c_w$
    $v_i(a_i = 1) = \min_{a' \in \{0,1\}} v_{i-1}(a_{i-1} = a') + (1 - a')c_t + c_w$
  end for
  % backward step
  $a_n^* = \arg\min_{a' \in \{0,1\}} v_n(a_n = a')$
  for $i = n - 1, ..., 1$ do
    $a_i^* = a_i(a_{i+1} = a_{i+1}^*)$
  end for
  return $\boldsymbol{a}^* = \{a_i^*\}_i$
```

The point is that the partial value $v_i(a_i)$ depends on the combination of actions $\{a_i\}_i$ only through the neighboring partial values $\{v_{i-1}(a_{i-1})\}_{a_{i-1}}$; namely,

$$\begin{aligned} v_i(a_i = 0) = \min_{a_{i-1}} \{ & v_{i-1}(a_{i-1} = 0) + q(s_i), \\ & v_{i-1}(a_{i-1} = 1) + q(s_i) \}, \\ v_i(a_i = 1) = \min_{a_{i-1}} \{ & v_{i-1}(a_{i-1} = 0) + c_t + c_w, \\ & v_{i-1}(a_{i-1} = 1) + c_w \}. \end{aligned}$$

This property means that we only have to calculate the partial values $\{v_i(a = 1), v_i(a = 0)\}_{i \in [n]}$ to get the optimal action $\boldsymbol{a}^*$, which takes only linear time with respect to the number of components n. The detailed algorithm is described in Algorithm 2.

4.3 Modeling q_i: the maintenance priority of i-th target

The state value $q_i = q(s_i; \boldsymbol{\theta})$ in (4) represents the priority (or the benefit) of performing maintenance for i-th component. Since the state cost $C_s(\boldsymbol{s})$ is non-decreasing in s_i, $q(s_i)$ should also be non-decreasing. In addition, components that maintenance performed are evaluated as zero except for maintenance costs, therefore the benefit for each component should be assigned non-negative values. Considering these properties, we employ the following non-negative and increasing parameterization for q.

$$q(s_i; \boldsymbol{\theta}) := \frac{\theta_3}{\theta_1} \log\left(1 + \exp(\theta_1(s_i - \theta_2))\right). \qquad (5)$$

The parameter θ_1 controls the smoothness. Since we adopt non-linear parameterization for q, convergence is not guaranteed [4]. Thus, we try several initial parameters for $\boldsymbol{\theta}$.

5. Experiment

We demonstrate the effectiveness of this approach with a simulated environment.

5.1 Environment settings

In the field of maintenance, there often is a variety in degradation rates of components, which is why CBM is employed instead

Table 1 The initial parameters tested

θ_0	{0.1, 1}
θ_1	{0.05, 0.1, 0.2, 0.5, 1, 2}
θ_2	{20, 25, 30, 35, 40, 45, 50}
θ_3	{0.1, 0.3, 1, 2}

of TBM. To reproduce these conditions, we employ the following transition models $\{M_i\}$ as the environment.

$$M_i(s,a) = \begin{cases} 1.1s + \epsilon\Delta_i & (a = 0) \\ 1.0 & (a = 1), \end{cases} \tag{6}$$

$$\epsilon \sim \exp(\mathcal{N}(0,1)), \tag{7}$$

where Δ_i is the characteristic degradation rate for i-th target, which is generated by sampling from $\Delta_i^{(base)} \sim \exp(\mathcal{N}(0, 1.3))$ and then applying the Gaussian filter ($std = 2$) for smoothness.

For the cost function, we employed the state cost function C_a in (3) with the parameters $\alpha = 50, c_s = 1$ and the action cost function C_s in (2) with the parameters $c_w = 2, c_t = 10$.

5.2 Training and testing settings

We set the number of targets $n = 1000$ and the training and the testing period $T_{train} = \{0, \dots, 1000\}, T_{test} = \{1001, \dots, 2000\}$, respectively. To generate the training data, we adopted the fixed section-based policy in (8) with the parameters $w = 10, \theta_t = 45$.

The random values Δ_i and $\epsilon_{t,i}$ are the same for all policies tested, i.e., CBM with various parameters and the proposed policy.

We set the discount parameter $\beta_{test} = 1$ for the test period. For the training period, too large discount parameter causes divergence, and thus we set $\beta_{train} = 0.9$.

We tried initial parameters of the Cartesian product of the candidate shown in Table 1 and selected the best parameter that minimizes the training objective $\sum_{t \in T_{train}} L_\theta$.

5.3 Baseline method: fixed section-based CBM

As the baseline method, the fixed section-based CBM approach (Fig. 1(b)) is examined. With the parameter of window width w, the targets are split into intervals in advance, and the action is taken for all targets in the section if the most degraded target in it is greater than the threshold θ_t.

$$\pi_{\text{CBM}}(a_i = 1|\boldsymbol{s}) = \begin{cases} 1 & \left(\max\limits_{j \in A_i}\{s_j\}_j \geq \theta_t\right) \\ 0 & \text{(otherwise)}, \end{cases} \tag{8}$$

where $A_i = \{j \mid \lfloor j/w \rfloor = \lfloor i/w \rfloor\}$ is the set of components that is in the same section with i-th component. The resulting condition history with parameters ($w = 10, \theta_t = 50$) is shown in Fig. 3, which is also used for generating training data. Performance under this policy is sensitive to the parameters as shown in Fig. 4. These parameters have to be optimized somehow using the training data, and which is another issue. To simplify the discussion, we used the optimal parameters selected by the test performance, and show our method with learned parameters still outperforms the baseline with the optimal parameters.

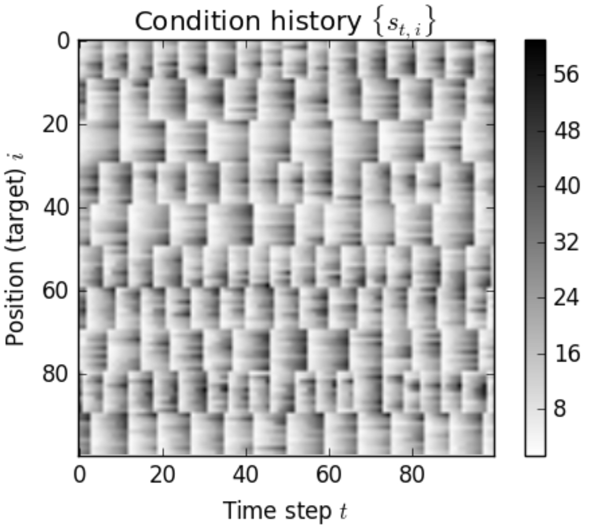


Fig. 3 State history $\{s_{t,i}\}_{t,i}$ under fixed section-based CBM policy (in Fig. 1(b) and (8)). Dark regions are degraded and thus need maintenance.

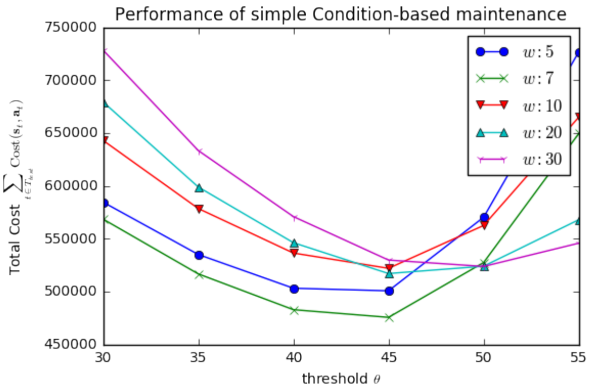


Fig. 4 Performance of fixed section-based maintenance policy in various hyperparameters. The performance is strongly dependent to the hyperparameters, the window width w and the threshold θ_t, and how to optimize them beforehand is not straightforward. Nonetheless, as a baseline method, we can assume these parameters are somehow optimized appropriately using the training data, thus we compare our method with the baseline method under the best hyperparameters in the test period ($w = 7, \theta_t = 45$).

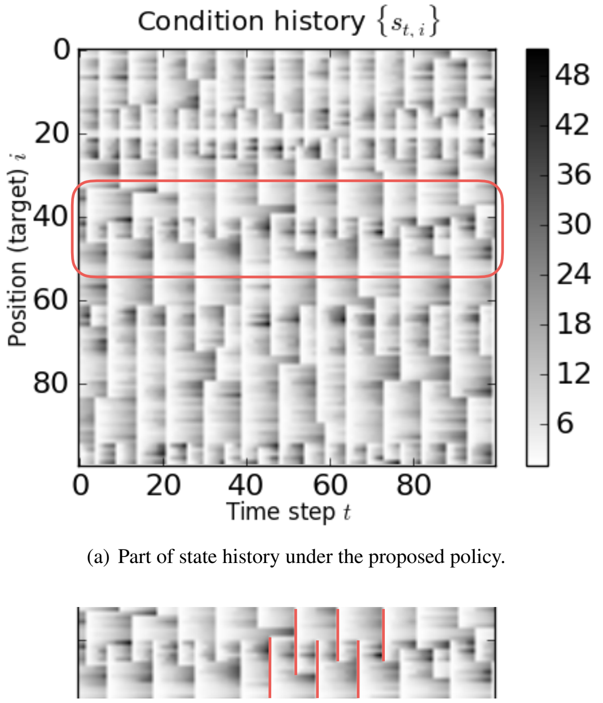
Table 2 Performance comparison

	Section-based CBM with best parameters	Proposed method
Total cost	4.76×10^5	$\mathbf{4.31 \times 10^5}$

6. Result and Discussion

The proposed method outperformed the baseline approach even when the best parameters (w, θ_t) in the test period are chosen for the baseline, as shown in Table 2.

A possible explanation of the performance of dynamic grouping is illustrated in Fig. 5. Rapidly degrading targets ($i \in [40, 45]$) are frequently maintained with negligible expense by selecting sections that cover such targets alternately (redlined in Fig. 5(b)). This alternate selection of sections cannot be realized in the fixed section-based approach, and we consider this is the benefit of the flexibility of dynamic grouping.



(a) Part of state history under the proposed policy.

(b) Framed section in (a) extracted.

Fig. 5 Condition history under dynamic grouping policy with learned parameters (a). The better performance of our approach (in Table 2) possibly comes from the exploitation of the local economic dependency and the variety of the degradation rates. Rapidly degrading targets (extracted in (b) are maintained frequently with the small number of groups by selecting groups alternately (indicated by red lines).

6.1 Interpretability in optimization

The advantage of the separability approximation of the state cost function, i.e., $C_s(\boldsymbol{s}) = \sum_i q_i(s_i)$, is not only the computational efficiency but also the interpretability in optimization. Black box optimization is hard to accept for maintenance decision makers in the field since they are responsible for the safety or have motivation for factors other than minimizing the explicitly defined cost function with observed data. As shown in Fig. 6, $q(s_i)$ can be interpreted as the maintenance priority of i-th target. We can plot them in the same graph with observed physical quantities, which maintainers are familiar with.

7. Conclusion

In this paper, we presented a condition-based infrastructure maintenance planning problem as a sequential and combinatorial optimization problem. This problem setting basically requires large-scale combinatorial optimization for the combination of current and future action of each component with considering the uncertainty in the future condition. To realize the dynamic grouping of small components of large infrastructures, we introduced local economic dependency assumption for maintenance cost. We proposed some approximations, namely, the Q-learning approach for temporal scalability and uncertainty, and a parameterized Q-function and dynamic programming for spatially scalable optimization of Q-function, which exploits the locality in economic dependency.

We demonstrated the performance in a simulated environment. The resulting condition history shows the advantage of dynamic grouping, that is, rapidly degrading targets can be maintained frequently by selecting alternate sections with the small extra expense only in working cost. The proposed method is not only better in performance but also interpretable, which we consider is also important for the maintenance decision makers to accept the recommended action. This is achieved by that the objective function of action optimization (Q-function) is separated into action cost and the sum of maintenance priority for each component, and which can be indicated in the same figure with the observed condition of each component. By comparing the cost and the maintenance priority, the maintenance planner can get to make a reasonable decision.

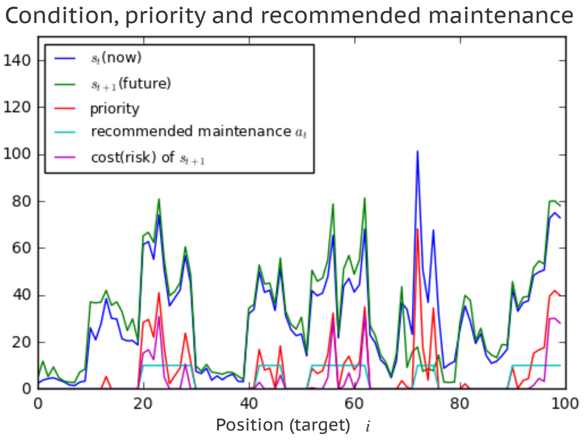


Fig. 6 A possible user interface of the maintenance recommendation system showing the current condition, current state cost, the estimated priority of maintenance, and the recommended maintenance groups. $q(s_i)$ can be interpreted as the maintenance priority of i-th target and we can plot them with observed physical quantities, which explains the reason for recommendation.

The following are some remaining issues or limitations, and possible extensions of our method. In real applications, historical data is sometimes limited. Since the transition of each target in one time step is summarized into one sample in our approach, our proposed method may be not sample-efficient. Thus, in those cases, we have to consider incorporating a model-based approach as in [5], in which the transition for each target is learned as a prediction model $\hat{M}(s_i, a_i)$. Also, in our experiment, we assumed that the condition observations are noise-free, but in the maintenance field, they often have severe noise or outliers. Therefore, estimating the true condition s_t, or calculating q_i from many observations (e.g., CNN-like model $q_i(s_{t-\tau:t,i-k:i+k})$) is an important possible extension. In addition, we focused on one-dimensional infrastructures. Other possible applications of the dynamic grouping approach include whole network setting such as NTD, and two-dimensionally distributed targets such as machine maintenance and inventory management of vending machines, ATMs and so on.

References

[1] A. H. Kobbacy, K. and N. P. Murthy, D.: *Complex System Maintenance Handbook* (2008).

[2] Chambon, S. and Moliard, J.-M.: Automatic road pavement assessment with image processing: review and comparison, *International Journal of Geophysics*, Vol. 2011 (2011).

[3] Dekker, R., Wildeman, R. E. and Van der Duyn Schouten, F. A.: A

review of multi-component maintenance models with economic dependence, *Mathematical Methods of Operations Research*, Vol. 45, No. 3, pp. 411–435 (1997).

[4] Getoor, L. and Taskar, B.: *Introduction to statistical relational learning*, MIT press (2007).

[5] Gu, S., Lillicrap, T., Sutskever, I. and Levine, S.: Continuous deep q-learning with model-based acceleration, *International Conference on Machine Learning*, pp. 2829–2838 (2016).

[6] Inaudi, D. and Glisic, B.: Long-range pipeline monitoring by distributed fiber optic sensing, *Journal of pressure vessel technology*, Vol. 132, No. 1, p. 011701 (2010).

[7] Jardine, A. K. and Tsang, A. H.: *Maintenance, replacement, and reliability: theory and applications*, CRC press (2005).

[8] Kim, S., Pakzad, S., Culler, D., Demmel, J., Fenves, G., Glaser, S. and Turon, M.: Health Monitoring of Civil Infrastructures Using Wireless Sensor Networks, *2007 6th International Symposium on Information Processing in Sensor Networks*, pp. 254–263 (online), DOI: 10.1109/IPSN.2007.4379685 (2007).

[9] Li, H.-N., Li, D.-S. and Song, G.-B.: Recent applications of fiber optic sensors to health monitoring in civil engineering, *Engineering structures*, Vol. 26, No. 11, pp. 1647–1657 (2004).

[10] Liu, J., Sun, L., Chen, W. and Xiong, H.: Rebalancing Bike Sharing Systems: A Multi-source Data Smart Optimization, *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ACM, pp. 1005–1014 (2016).

[11] Nguyen, K.-A., Do, P. and Grall, A.: Multi-level predictive maintenance for multi-component systems, *Reliability engineering & system safety*, Vol. 144, pp. 83–94 (2015).

[12] Nicolai, R. P. and Dekker, R.: Optimal maintenance of multi-component systems: a review, *Complex system maintenance handbook*, Springer, pp. 263–286 (2008).

[13] Papadakis, I. S. and Kleindorfer, P. R.: Optimizing infrastructure network maintenance when benefits are interdependent, *OR Spectrum*, Vol. 27, No. 1, pp. 63–84 (2005).

[14] Riedmiller, M.: Neural fitted Q iteration–first experiences with a data efficient neural reinforcement learning method, *European Conference on Machine Learning*, Springer, pp. 317–328 (2005).

[15] Tian, Z. and Liao, H.: Condition based maintenance optimization for multi-component systems using proportional hazards model, *Reliability Engineering & System Safety*, Vol. 96, No. 5, pp. 581–589 (2011).

[16] Van Horenbeek, A. and Pintelon, L.: A dynamic predictive maintenance policy for complex multi-component systems, *Reliability Engineering & System Safety*, Vol. 120, pp. 39–50 (2013).

[17] Watkins, C. J. and Dayan, P.: Q-learning, *Machine learning*, Vol. 8, No. 3-4, pp. 279–292 (1992).