Distributed optimization for railway track maintenance operations planning

Faris, Muhammad; Núñez, Alfredo; Su, Zhou; De Schutter, Bart

DOI
10.1109/ITSC.2018.8569335

Publication date
2018

Document Version
Final published version

Published in
Proceedings of the 2018 IEEE Intelligent Transportation Systems Conference (ITSC 2018)

Citation (APA)
Faris, M., Núñez, A., Su, Z., & De Schutter, B. (2018). Distributed optimization for railway track maintenance operations planning. In *Proceedings of the 2018 IEEE Intelligent Transportation Systems Conference (ITSC 2018)* (pp. 1194-1201). Article 8569335 IEEE. https://doi.org/10.1109/ITSC.2018.8569335

2018 21st International Conference on Intelligent Transportation Systems (ITSC)
Maui, Hawaii, USA, November 4-7, 2018

Distributed Optimization for Railway Track Maintenance Operations Planning

Muhammad Faris
Delft Center for Systems and Control
Delft University of Technology
Delft, the Netherlands
mhmmdfrs@gmail.com

Alfredo Núñez
Section of Railway Engineering
Delft University of Technology
Delft, the Netherlands
a.a.nunezvicencio@tudelft.nl

Zhou Su
Delft Center for Systems and Control
Delft University of Technology
Delft, the Netherlands
z.su-1@tudelft.nl

Bart De Schutter
Delft Center for Systems and Control
Delft University of Technology
Delft, the Netherlands
b.deschutter@tudelft.nl

***Abstract*—In this paper, distributed optimization approaches are developed for the planning of maintenance operations of large-scale railway infrastructure formulated as a Mixed-Integer Linear Programming (MILP) problem. The proposed planning problem is solved using three different distributed optimization schemes: Parallel Augmented Lagrangian Relaxation (PALR), Alternating Direction Method of Multipliers (ADMM), and Distributed Robust Safe But Knowledgeable (DRSBK). The original distributed algorithms are modified to handle the non-convex nature of the optimization problem and to improve the solution quality. The results of large-scale test instances show that DRSBK can outperform the other distributed approaches, by providing the closest-to-optimum solution while requiring the lowest computation time.**

***Keywords*—track maintenance planning, railway engineering, mixed-integer programming, distributed optimization**

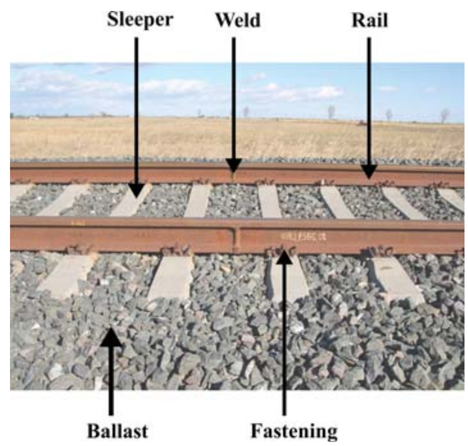

Fig. 1: Components of a railway track

I. Introduction

Railway infrastructure consists of different assets, comprising rails, sleepers, fastenings, welds, ballast, and so forth, as depicted in Figure 1. All assets are interconnected and work together. Among them, the ballast is a vital component as it is used to support the track level and to regulate the alignment at the designated positions [1]. Due to regular usage of tracks, the quality of ballast degrades over time. In order to control the degradation, ballast must be maintained so that its performance can always meet the technical and safety standards. Maintenance actions for ballast, such as tamping, cleaning, or renewal must be decided by infrastructure managers, including the time, location, and type of intervention.

Maintenance schedules are not always easy to obtain systematically. Cost and track performance should have to be optimized subject to multiple constraints. In the literature, decision support systems have been proposed to process different sources of railway track condition data and to determine (near) optimal schedules. In [2], the planning of tamping operations over a whole railway track is developed. The goal is to minimize the total tamping costs. In [3], renewal operations are considered alongside tamping. A maintenance model is used in [4] to take into account different construction works in a large-scale railway infrastructure.

It is crucial for the real-life deployment of a decision support system to acknowledge that a railway track consists of a large number of sections. Moreover, in the case of ballast maintenance, the railway industry requires decisions tailored to specific locations, meaning that local track section models should be less aggregated over space than current available models in the literature, moving from models for 1km, 100m towards 10m of track. If each track section has independent dynamics, optimal railway track maintenance decision making will result in a large-scale optimization problem with a huge number of decision variables. In the literature, approaches are mostly based on centralized optimization schemes, where the information processing and computation of all decision variables are conducted in a single centralized node [2, 3, 4, 5, 6]. From a computational perspective, the scheme is unattractive when the number of track sections and thus also the number of decision variables increase as a linear increase in the number of decision variables in such maintenance optimization problems implies an exponential increase in the required computation time [7].

In [8], a multi-level optimization scheme is proposed for railway maintenance. The proposed approach is split over high-level, middle-level, and low-level optimizers, allowing to separate the computation process at different levels. A hierarchical-based optimization for railway maintenance of different components has also been proposed in [9]. However, the hierarchical schemes are highly dependent on the primary structure of their respective problems and thus they become less flexible to solve large-scale problems.

In general, optimization of maintenance planning can be formulated as a mixed-integer programming (MIP) problem. In the literature, various Lagrangian-based distributed approaches were successfully developed to solve nonlinear real-valued problems and then applied to mixed-integer problems, such as the augmented Lagrangian method and Alternating Direction Method of Multipliers (ADMM). For instance, results in [10, 11] demonstrate their performance. In [12], a continuous relaxation technique for the binary variables is used. The resulting solution yields a bound to the original objective function and provides a warm start of the optimization of the mixed-integer programming problem. Apart from Lagrangian-based methods, an algorithm called Distributed Robust Safe But Knowledgeable (DRSBK) [13] also has been applied to a mixed-integer programming problem with hard non-convex coupling. This algorithm utilizes a coupling tightening approach when solving one subproblem. In this way, the couplings can be decomposed and solved individually. The distributed approaches discussed can be seen as heuristic methods to solve the MIP problem because there is no guarantee for convergence toward the global optimum [14].

In this paper, the main contribution is in applying distributed optimization approaches to address large-scale railway track maintenance operation planning problems. The focus is on reducing the computation time and increasing the scalability of the optimization method to handle large-scale instances while obtaining good solutions. To that end, two Lagrangian-based decomposition approaches and a constraint tightening distributed optimization approach (DRSBK) are implemented. The centralized formulation and solution of the problem was proposed in [3]. In this paper the focus is on the distributed optimization approaches.

This paper is organized as follows. The maintenance optimization problem is firstly described in Section II. Section III addresses the development of distributed optimization approaches. Then, two case studies are discussed in Section IV. Finally, Section V provides the conclusions and future work of this research.

II. Problem definition

The formulation is based on the centralized approach we proposed in [3]. Besides, some constraints are added to the optimization problem to incorporate practical issues such as maintenance closure times, budget limitations, and early renewal prevention. When it is installed for the first time, the ballast has sharp-edged stones that form a rigid foundation for the rail. As the tracks are regularly used by rolling stock, the ballast is misplaced gradually, and its condition deteriorates over time. In the railway industry, regular maintenance interventions are performed to improve the track performance, including tamping and renewal.

Tamping is performed to regulate the track alignment so that the track performance is improved to a certain level. When tamping is no longer effective, a renewal operation is undertaken to completely replace the old ballast. The dynamics of ballast degradation and the corresponding maintenance options can be mathematically modeled as follows:

$$\begin{aligned} x_{1,i}(k+1) &= a_{1,i}x_{1,i}(k) + f_{1,i}(x_i(k), u_i(k)) \\ x_{2,i}(k+1) &= a_{2,i}x_{2,i}(k) + f_{2,i}(x_i(k), u_i(k)) \end{aligned} \tag{1}$$

$$x_i(k) = \begin{bmatrix} x_{1,i}(k) & x_{2,i}(k) \end{bmatrix}^T$$

where the state variables $x_{1,i}(k)$ and $x_{2,i}(k)$ are the track performance level and the degradation memory at time step k for track section i, respectively. Memory is a variable considered to account for the fact that maintenance actions will not be able to reach a track condition as in the case of newly installed/replaced track. Moreover, $a_{1,i}$ is the track degradation rate and $a_{2,i}$ is the memory evolution rate. The degradation process is largely determined by these parameters and their values can be different for each track section. The railway track is a distributed parameter system that changes over time and over space. The functions $f_{1,i}$ and $f_{2,i}$ are discontinuous functions defined as:

$$f_{1,i}(x_i(k), u_i(k)) = \begin{cases} 0, & \text{if } u_i(k) = 1 \\ -a_{1,i}x_{1,i}(k) + \\ \qquad a_{2,i}x_{2,i}(k) + \alpha, & \text{if } u_i(k) = 2 \\ -a_{1,i}x_{1,i}(k) + h_{\text{min}}, & \text{if } u_i(k) = 3 \end{cases}$$

and

$$f_{2,i}(x_i(k), u_i(k)) = \begin{cases} 0, & \text{if } u_i(k) = 1 \\ \alpha, & \text{if } u_i(k) = 2 \\ -a_{2,i}x_{2,i}(k) + h_{\text{min}}, & \text{if } u_i(k) = 3 \end{cases}$$

where α and h_{min} are the tamping offset and minimum degradation threshold. This system has three types of inputs to decide at each time step: no maintenance $u_i(k) = 1$, tamping $u_i(k) = 2$, or renewal $u_i(k) = 3$. Note that the model proposed here has slight differences from the one in [3]. First, the memory variable is set to have a rate $a_{2,i}$. The memory is assumed to evolve exponentially over time. This assumption is different from [3], where a linear evolution of the memory was considered.

Due to the use of integer inputs, this system is basically non-linear. One way to incorporate integer maintenance decisions into the degradation dynamics is by using the Mixed Logical Dynamical (MLD) framework. Three options of maintenance input can be represented by two binary variables $\delta_{1,i}(k)$ and $\delta_{2,i}(k)$ for renewal and tamping respectively at track section i. The conversion table is given in Table I.

TABLE I: Conversion of system input and binary variables

$u_i(k)$	$\delta_{1,i}(k)$	$\delta_{2,i}(k)$
1	0	0
2	0	1
3	1	0

The option to perform both tamping and renewal at the same time step k for the same section i is eliminated by using the following constraint:

$$\delta_{1,i}(k) + \delta_{2,i}(k) \leq 1 \tag{2}$$

By taking into account the binary variables, the non-linear model is then formulated as follows:

$$\begin{aligned} x_{1,i}(k+1) &= a_{1,i}x_{1,i}(k) + \delta_{1,i}(k)(-a_{1,i}x_{1,i}(k) + h_{\min}) + \\ &\quad \delta_{2,i}(k)(-a_{1,i}x_{1,i}(k) + a_{2,i}x_{2,i}(k) + \alpha) \\ x_{2,i}(k+1) &= a_{2,i}x_{2,i}(k) + \delta_{2,i}(k)\alpha + \delta_{1,i}(k)(-a_{2,i}x_{2,i}(k) + \\ &\quad h_{\min}) \end{aligned}$$

Auxiliary variables are introduced and the model for track section i can then be written in the following linear matrix form:

$$\begin{bmatrix} x_{1,i}(k+1) \\ x_{2,i}(k+1) \end{bmatrix} = \begin{bmatrix} a_{1,i} & 0 \\ 0 & a_{2,i} \end{bmatrix} \begin{bmatrix} x_{1,i}(k) \\ x_{2,i}(k) \end{bmatrix} + \begin{bmatrix} h_{\min} & \alpha \\ h_{\min} & \alpha \end{bmatrix} \begin{bmatrix} \delta_{1,i}(k) \\ \delta_{2,i}(k) \end{bmatrix} + \begin{bmatrix} -a_{1,i} & -a_{1,i} & a_{2,i} & 0 \\ 0 & 0 & 0 & -a_{2,i} \end{bmatrix} \begin{bmatrix} z_{1,i}(k) \\ z_{2,i}(k) \\ z_{3,i}(k) \\ z_{4,i}(k) \end{bmatrix} \tag{3}$$

where the auxiliary variables are defined as:

$$\begin{aligned} z_{1,i}(k) &= \delta_{1,i}(k)x_{1,i}(k) & z_{2,i}(k) &= \delta_{2,i}(k)x_{1,i}(k) \\ z_{3,i}(k) &= \delta_{2,i}(k)x_{2,i}(k) & z_{4,i}(k) &= \delta_{1,i}(k)x_{2,i}(k) \end{aligned} \tag{4}$$

and the auxiliary vector can be defined as $z_i(k) = \begin{bmatrix} z_{1,i}(k) & z_{2,i}(k) & z_{3,i}(k) & z_{4,i}(k) \end{bmatrix}^T$. The conditions (4) can be reformulated using the following four linear constraints [15]:

$$\begin{aligned} z_{p,i}(k) &\leq h_{\max}\delta_{l,i}(k) \\ z_{p,i}(k) &\geq h_{\min}\delta_{l,i}(k) \\ z_{p,i}(k) &\leq x_{j,i}(k) - h_{\min}(1 - \delta_{l,i}(k)) \\ z_{p,i}(k) &\geq x_{j,i}(k) - h_{\max}(1 - \delta_{l,i}(k)) \end{aligned} \tag{5}$$

for $p \in \{1,2,3,4\}$ and $l \in \{1,2\}$ where $h_{\max}$ is the maximum degradation threshold. In this way, there will be sixteen linear equations for each track section.

Let $V_i(k)$ contain the following binary and auxiliary variables:

$$V_i(k) = \begin{bmatrix} \delta_{1,i}(k) & \delta_{2,i}(k) & z_i(k) \end{bmatrix}^T.$$

Hence, the reformulated state space model is of the form:

$$x_i(k+1) = A_i x_i(k) + B_i V_i(k) \tag{6}$$

where A_i and B_i are matrices that can be obtained from [3].

The initial condition and degradation rate for each track section are defined according to various case studies to capture the distributed dynamics of the railway tracks. The degradation rate is assumed to be constant. The corresponding constraints include the initial condition:

$$x_{1,i}(0) = x_{i,0}^1, \quad x_{2,i}(0) = x_{i,0}^2 \tag{7}$$

where $x_{i,0}^1$ and $x_{i,0}^2$ are the initial values taken from measurements for state variable degradation level and offset memory, respectively. Next, the following constraint is essential to ensure that the track performance is always within an acceptable range:

$$h_{\min} \leq x_{1,i}(k) \leq h_{\max}, \quad h_{\min} \leq x_{2,i}(k) \leq h_{\max} \tag{8}$$

A renewal operation can be allowed only once the offset memory is considered high. This leads to the following conditional constraint:

$$x_{2,i}(k) - h_{\mathrm{r}} \geq (r_i - 1)h_{\max} \tag{9}$$

$$r_i - \delta_1(k) \geq 0 \tag{10}$$

where r_i and h_{r} are the switching binary indicator at track section i and prevention threshold for allowing renewal operation, respectively. Another constraint is that the maintenance budget is limited [16]. The following constraints consider that the number of interventions, both tamping and renewal, over the prediction horizon is restricted by the thresholds:

$$\sum_{k=1}^{T} \delta_{2,i}(k) \leq g_{\mathrm{t}} \tag{11}$$

$$\sum_{k=1}^{T} \delta_{1,i}(k) \leq g_{\mathrm{r}} \tag{12}$$

where g_{t} and g_{r} are the maximum numbers of allowed tamping and renewal operations over the prediction horizon T, respectively.

The previously defined constraints can be categorized as individual constraints, which only affect their respective track section i. Alongside them, coupling constraints, which influence multiple track sections, exist. One of them involves the maintenance closure time. In densely used railways, the maintenance time slot is usually less than 7 hours and the maintenance operations are only allowed during night time at weekends [1]. The maximum allowed closure time is hence denoted as $t_{\max}$. This constraint applies for both tamping and renewal, respectively. Based on [5], [8], this constraint can be written as follows:

$$\sum_{i \in P(k)} t_{\mathrm{t1}}\delta_{2,i}(k) + t_{\mathrm{t2}}(1 - \delta_{2,i}(k)) \leq t_{\max} \tag{13}$$

$$\sum_{i \in P(k)} t_{\mathrm{r1}}\delta_{1,i}(k) + t_{\mathrm{r2}}(1 - \delta_{1,i}(k)) \leq t_{\max} \tag{14}$$

where t_{t1} and t_{t2} are maintenance operation time and traveling time for the tamping machines to reach the track section i from a certain position, respectively. The set $P(k)$ is the set of track sections that the machines (for tamping and renewal) will maintain or pass over during maintenance operation at time step k. The same representations also hold for t_{r1} and t_{r2} for renewal, with different time values. Moreover, it is assumed that the machines move in one direction at each time step, from a starting point toward an endpoint at the other end of the track. Since constraints (13) and (14) affect the maintenance schedule across multiple track sections, they can be considered as coupling constraints.

The optimal state variables $\bar{X}$ and decision variables $\bar{V}$ for all track sections over the prediction horizon can be obtained through minimizing the following objective function:

$$\begin{aligned} J(\bar{X},\bar{V}) &= \sum_{i=1}^{N}\sum_{k=1}^{T} Qx_i(k) + \lambda RV_i(k) \\ &= \sum_{i=1}^{N} J_{\text{ind}}(\tilde{X}_i,\tilde{V}_i) \end{aligned} \tag{15}$$

where Q and R are matrices with only non-negative entries, and $\tilde{X}_i$ and $\tilde{V}_i$ are state and input variables over the prediction horizon for the track section i. The state variables $x_i(k)$ in (15) can be substituted using the same technique as in [3], leaving the $V_i(k)$ as the only decision variable. Furthermore, the optimization problem can be written in the following form:

$$\underset{\bar{V}}{\text{minimize}} \quad J(\bar{V}) = \sum_{i=1}^{N} J_{\text{ind}}(\tilde{V}_i) \tag{16}$$

$$\begin{aligned} \text{subject to} \quad & E\bar{V} \leq g_{\text{ind}} \\ & \sum_{i=1}^{N} F_i\tilde{V}_i \leq g_{\text{coup}} \end{aligned} \tag{17}$$

where E and g_{ind} are the parameter matrix and vector associated with all individual constraints, respectively, while F_i and g_{coup} are the parameter matrix and vector associated with the coupling constraints. This explicit separation is useful for the implementation of distributed approaches. Moreover, there is no coupling variable between subproblems in the objective function. Besides, it is noteworthy that the optimization problem can be categorized as NP-hard, Mixed-Integer Linear Programming (MILP) due to the use of both continuous and discrete variables.

III. Distributed Optimization

Three distributed optimization approaches are implemented in this work: Parallel Augmented Lagrangian Relaxation (PALR), Alternating Direction Method of Multipliers (ADMM), and Distributed Robust Safe But Knowledgeable (DRSBK). Further explanations about each approach are given below.

A. Parallel Augmented Lagrangian Relaxation (PALR)

In order to cope with the requirement for implementing PALR, the centralized problem in (16) and (17) has to be transformed into an augmented Lagrangian form [17]. Another requirement is that any inequality coupling constraint in the proposed problem must be converted into the equality form [12]. Thus, a vector $\bar{S}$ is defined as a slack variable for the couplings, for both tamping and renewal, over the prediction horizon. The augmented Lagrangian can be written as follows:

$$\begin{aligned} L(\bar{V},\bar{S},\gamma) = &\sum_{i=1}^{N} J_{\text{ind}}(\tilde{V}_i) + \sum_{k=1}^{T} s(k) + \\ &\gamma\left(\sum_{i=1}^{N} F_{\text{i}}\tilde{V}_i + \sum_{k=1}^{T} F_{\text{s}}s(k) - g_{\text{coup}}\right) + \\ &\frac{\rho}{2}\left\|\sum_{i=1}^{N} F_{\text{i}}\tilde{V}_i + \sum_{k=1}^{T} F_{\text{s}}s(k) - g_{\text{coup}}\right\|_2^2 \end{aligned} \tag{18}$$

$$\text{subject to} \quad E\bar{V} \leq g_{\text{ind}} \tag{19}$$

where F_{i} and F_{s} are the parameter matrices of the coupling constraints for the input and slack variables. Moreover, $s(k)$ is a slack vector for the time step k and so $\bar{S} = \begin{bmatrix} s^T(1) & \ldots & s^T(T) \end{bmatrix}^T$. Furthermore, the Lagrangian equation for the dual problem can be written as:

$$q(\gamma) = \inf_{\bar{V},\bar{S}} \left(L(\bar{V},\bar{S},\gamma)\right) \tag{20}$$

Then, the dual variables can be updated by solving the maximization problem of the above function [18]. A subgradient method is in practice used to perform the dual update.

In each iteration, all subproblems can be solved in parallel. Once all subproblems have been solved, the results are collected by a coordinator to be included in the update of the dual variable. The presence of the coordinator also implies that one dual variable γ is used to determine the common price for all subproblems.

B. Alternating Direction Method of Multipliers (ADMM)

Basically, ADMM shares the augmented Lagrangian equation with PALR. The difference is in the way of decomposing the quadratic terms of the Lagrangian equation. Instead of linearizing these quadratic terms, ADMM uses the so-called alternating technique. This technique enables the separation of the quadratic terms to be determined individually by fixing the decisions coming from the other subproblems [19]. This also implies that the approach runs in sequence. In this way, ADMM can exploit the latest decisions from the other subproblems. The unscaled form of ADMM [14] is chosen for the ease of implementation.

C. Modifications for the Lagrangian-based approaches

To deal with the proposed problem, some modifications of the original PALR and ADMM approaches are required. The non-convex non-smooth nature of coupling constraints (13) and (14) is due to all the discrete decision variables involved in the constraints. Recursive gradient-based methods will be alternatingly jumping between these discrete points. Consequently, the subgradient dual update iterations might be unable to converge or even drive toward a feasible region. One way to solve this problem is by applying a continuous relaxation technique to the binary decision variables, such that the MILP becomes a less complex Linear Programming (LP) problem [10]. On top of that, the objective function value obtained from solving the LP problem can be used as a lower bound for the MILP optimization in the next step:

$$J_{\text{ind}}(\tilde{V}_i^{\text{LP}}) \leq J_{\text{ind}}(\tilde{V}_i)$$

where $\tilde{V}_i^{\text{LP}}$ is the LP form of the decision vector. The decision variables can also be used as a warm start vector. This two-step process is performed in each iteration, in both PALR and ADMM.

Since the convergence of the primal residual cannot be guaranteed, the implemented algorithm obtains the best objective value found during the whole iteration procedure. Furthermore, to terminate the iterations, two requirements must be fulfilled. First, the residual vector of couplings is checked in each iteration j to assess whether the feasibility conditions have been reached [12]. The residual vector at iteration j, $r^{(j)}$, is defined as follows:

$$r^{(j)} = g_{\text{coup}} - \left(\sum_{i=1}^{N} F_{\text{i}} \tilde{V}_i^{(j)} + \sum_{k=1}^{T} F_{\text{s}} s(k)^{(j)} \right).$$

The feasibility conditions are indicated by non-positive values on the entire rows of the vector or $r^{(j)} \leq 0$. Therefore, the maximum closure times over the prediction horizon are always satisfied. Next, if the difference between the objective function value of current iteration and the best found objective function value is below the threshold, the iteration is terminated.

D. Distributed Robust Safe but Knowledgeable (DRSBK)

In [20], DRSBK is applied to the distributed optimization of multi-vehicle or multi-agent coordination. This algorithm is originally devoted to MILP problems. The concept is as follows: instead of including the coupling constraints into the Lagrangian form objective function, this algorithm applies tightening resource allocation in the coupling constraints for each subproblem. This can be illustrated by the following expressions:

$$\min_{\tilde{V}_i} \quad J_{\text{ind}}(\tilde{V}_i) \tag{21}$$

subject to (19)

$$F_i \tilde{V}_i \leq g_{\text{coup}} - \sum_{j=1, j \neq i}^{N-1} F_j \tilde{V}_j \tag{22}$$

where $\tilde{V}_i$ contains the decision variables for track section i over the prediction horizon. The second set of constraints in (22) are the couplings with reduced resources, which are the remaining available maintenance closure times over the prediction horizon. The reduction process can be done by fixing the decisions from other subproblems. In this way, the coupling constraints can be decoupled and so the problem can be solved individually in a sequential and non-iterative way. One advantage of assigning the couplings into individual constraints is that the feasible solutions are much easier to retrieve.

Moreover, unlike the coordinator in Lagrangian-based methods, the task of the coordinator in DRSBK is only checking the feasibility of the generated solution.

E. Random Sequence Generator and Stopping Criterion

The solution of the original non-iterative version of DRSBK might get stuck in a local optimum or even become infeasible without any chance of updating the solution. Hence, the algorithm is modified in such a way that the sequence of subproblems to be processed in each iteration is generated randomly. If the output from the solver indicates that the result from an iteration is not feasible, the sequence is generated again randomly. The feasibility checking technique is different from the Lagrangian-based algorithms, in the sense that it sums up the individual feasible indicators given by all subproblems. The result is feasible if the total indicator value is equal to the number of sections. Moreover, the stopping criterion is similar to that of the Lagrangian-based algorithms.

IV. CASE STUDIES

In this section, three distributed optimization approaches (PALR, ADMM, and DRSBK) are compared and analyzed. All simulations in this research are conducted on a general purpose computer with an Intel Core-i5 processor and 8GB of RAM. All the LP and MILP problems are solved by the Gurobi optimizer 7.5, called from MATLAB R2017a. Moreover, the following assumptions and general settings are considered:

- The time step for maintenance intervention is one month. The control horizon corresponds to consecutive six months, according to the Dutch railway case study. The prediction horizon is set to be nine months.
- Parameters (initial condition and degradation rate) for each track section can be different, according to corresponding scenarios. One scenario typically presents different parameters for each track section, which is randomly generated as a Gaussian distribution over locations. In this way, the spatial correlation between different track sections is included while also considering that some track sections have faster rates than others [21, 5]. Furthermore, different scenarios are simulated by using non-uniform various Gaussian settings, with $\mu \in [1.012, 1.050]$ and $\sigma \in [0.07, 0.08]$. Also, the degradation rate is assumed to be known and constant within the simulation horizon.
- The model is deterministic, meaning that no stochasticity or any perturbation from, for instance, reactive maintenance is involved. Moreover, the trade-off weight in (15) is chosen to be $\lambda = 10$.
- Maintenance machine and personnel are assumed to be always available. The case study involves a single railway track, consisting of a number of track sections. Each track section length is 200 m. The layout is depicted in Figure 2.

A. Experiment 1: Test on a 150 track sections case

In this experiment, each approach is tested on a case with 150 track sections. In this case, the number of variables is considered large enough to make the centralized approach to be no longer tractable. The comparison for each approach is presented in Table II. The table presents the average values from ten different scenarios.

Among the distributed approaches, DRSBK is the fastest. This is due to the modification of the resource allocation instead of the augmented Lagrangian. Additionally, its solutions are closer to the global optimum compared to PALR and ADMM. Consequently, the solutions given by the Lagrangian-based algorithms are suboptimal as no convergence guarantee to the solutions of the non-convex problem. The gap from each approach with respect to the global optimum can be further observed in Table II. Furthermore, DRSBK suggests more tamping operations than the centralized approach. The same number of renewal operations is suggested. On the other hand, the two Lagrangian-based approaches suggest a higher number of tamping and renewal operations than the centralized approach, which is not preferable from an economical perspective.

The evolution of one track degradation curve and the decision plots generated by all approaches are depicted in Figure

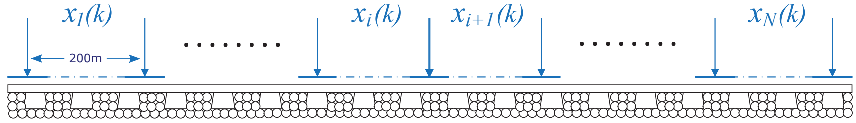

Fig. 2: Illustration of track sections

3. The first remark is that all approaches successfully maintain the degradation level within the safety bounds. However, they suggest different numbers of tamping operations at different times. This can be seen by the decisions given by PALR and ADMM, as they perform the first tamping earlier than in the centralized approach. Moreover, DRSBK suggests in the example the same maintenance decisions for the shown track section as the centralized approach.

All distributed approaches have a significantly faster processing time than the centralized approach. To conclude, DRSBK outperforms PALR and ADMM, both from the maintenance performance operational case and computational point of view.

B. Experiment 2: Gradual increase in the number of track sections

The second experiment presents a comparison with a gradual increase in the number of track sections. Once a computation time reaches the threshold for the centralized approach (in this experiment, it is 1100 seconds), the corresponding approach is no longer considered.

The simulation results of the first criterion (computation time) against the number of track sections are depicted in Figure 4a. It can be observed that the computation time increases with the number of track sections. This issue is experienced not only by the centralized approach but also the distributed approaches. However, the curve of the centralized approach is exponential, which can be expected for such an NP-hard problem. The centralized approach stops the experiment earlier than the other algorithms, at $N = 150$. PALR can continue to perform computationally reasonably up to 400 track sections. ADMM can prolong until around 900 track sections. ADMM outperforms PALR possibly due to the use of the newest instead of the previous iteration data. On top of that, DRSBK can deal with up to 1300 track sections. Thus, DRSBK is the most scalable approach.

Furthermore, Figure 4b indicates that the DRSBK solutions are better compared to the other two Lagrangian-based approaches solutions with increasing number of sections test.

V. Conclusions

Three different distributed optimization approaches have been tested for large-scale railway track maintenance operations planning. The first two distributed optimization approaches work based on Lagrangian duality theory: Parallel Augmented Lagrangian Relaxation (PALR) and Alternating Direction Method of Multipliers (ADMM). To drive the iterations toward feasible regions, PALR and ADMM are modified using a two-step process (LP and MILP). Alongside the Lagrangian-based approaches, Distributed Robust Safe But Knowledgeable (DRSBK) is implemented. To avoid infeasible solutions and to obtain more optimal solutions while maintaining simplicity, the algorithm is extended to be iterative, and the sequence of processed subproblems is generated randomly.

In the case studies, it is shown that the distributed optimization approaches can solve the proposed problem quicker than the centralized approach with the number of track sections above 150. DRSBK is the fastest approach, and it is also able to generate the closest solution to the centralized one. Furthermore, ADMM is quicker than PALR.

As part of further research, the use of real-life data to include stochastic degradation behavior into the model can also be utilized to consider uncertainties in railway maintenance operations. Additionally, the current scheme can be extended into a distributed hierarchical one to facilitate different time scales in maintenance planning.

References

[1] C. Esveld. *Modern Railway Track*. MRT-Productions, 2001.

[2] M. Wen, R. Li, and K.B. Salling. "Optimization of preventive condition-based tamping for railway tracks". *European Journal of Operational Research* 252.2 (2016), pp. 455–465.

[3] Z. Su, A. Núñez, A. Jamshidi, S. Baldi, Z. Li, R. Dollevoet, and B. De Schutter. "Model predictive control for maintenance operations planning of railway infrastructures". *Proceedings of Computational Logistics*. Springer International Publishing, 2015, pp. 673–688.

[4] R. Li and R. Roberti. "Optimal scheduling of railway track possessions in large-scale projects with multiple construction works". *Journal of Construction Engineering and Management* 143.6 (2017).

[5] S. M. Famurewa, T. Xin, M. Rantatalo, and U. Kumar. "Optimisation of maintenance track possession time: a tamping case study". *Proceedings of the Institution of Mechanical Engineers, Part F: Journal of Rail and Rapid Transit* 229.1 (2015), pp. 12–22.

[6] E. Gustavsson. "Scheduling tamping operations on railway tracks using mixed integer linear programming". *EURO Journal on Transportation and Logistics* 4.1 (2015), pp. 97–112.

[7] F. Guo, C. Wen, J. Mao, G. Li, and Y. D. Song. "A distributed hierarchical algorithm for multi-cluster constrained optimization". *Automatica* 77 (2017), pp. 230–238.

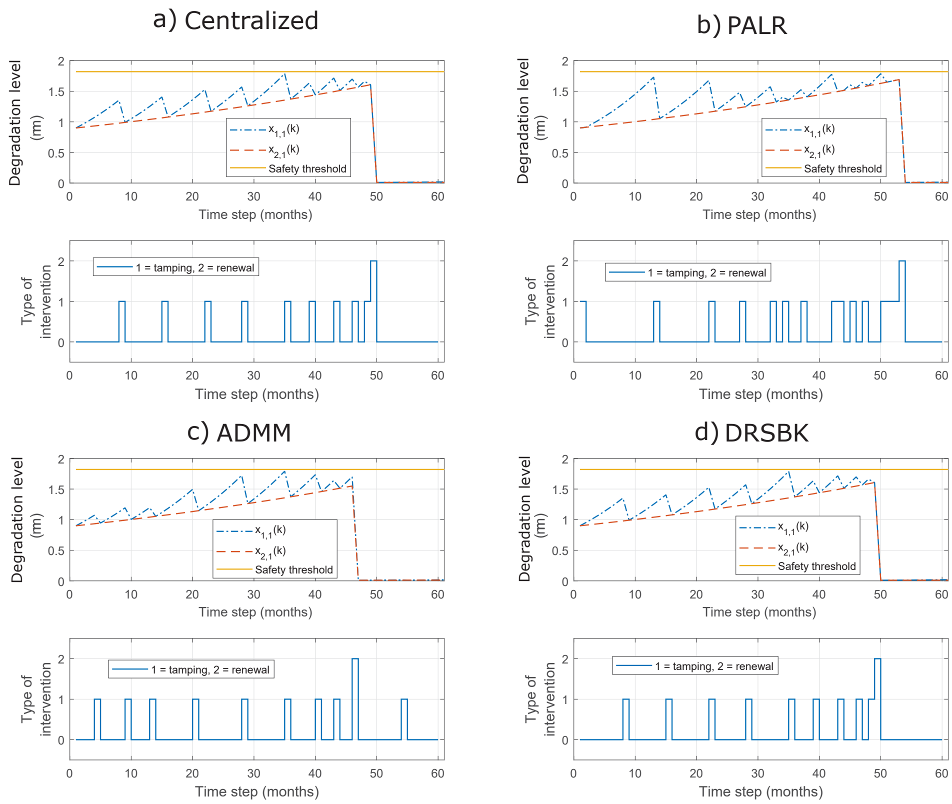

Fig. 3: Degradation and decision curves: a) Centralized, b) PALR, c) ADMM, and d) DRSBK

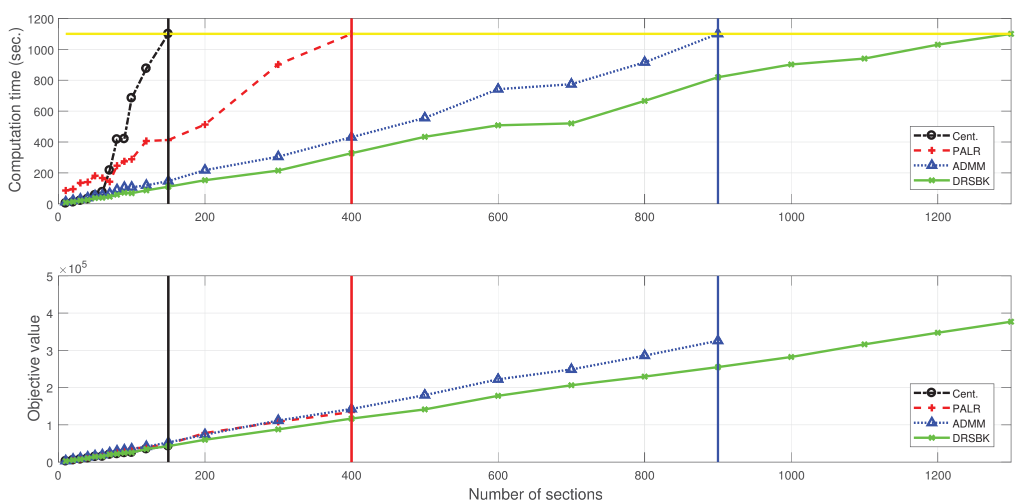

Fig. 4: Comparison of a) computation time and b) objective function value of centralized and and distributed approaches

TABLE II: Performance comparison for $N = 150$

Parameter / algorithm	Centralized	PALR	ADMM	DRSBK
Total performance	36200.18	54137.37	45174	36205.13
Computation time (second)	871.84	453.74	160.12	104.67
Number of tamping operations	493.00	688.00	946.00	493.5
Number of renewal operations	43.5	94.78	58.30	43.5
Track performance	18220.18	18824.03	15300.91	18220.13
Normalized performance	-	-52.00%	-29.00%	-0.02%

[8] Z. Su, A. Jamshidi, A. Núñez, S. Baldi, and B. De Schutter. "Multi-level condition-based maintenance planning for railway infrastructures – A scenario-based chance-constrained approach". *Transportation Research Part C: Emerging Technologies* 84 (2017), pp. 92–123.

[9] K. Verbert, B. De Schutter, and R. Babuška. "Timely condition-based maintenance planning for multi-component systems". *Reliability Engineering & System Safety* 159 (2017), pp. 310–321.

[10] M. J. Feizollahi, M. Costley, S. Ahmed, and S. Grijalva. "Large-scale decentralized unit commitment". *International Journal of Electrical Power and Energy Systems* 73 (2015), pp. 97–106.

[11] C. He, L. Wu, T. Liu, and M. Shahidehpour. "Robust co-optimization scheduling of electricity and natural gas systems via ADMM". *IEEE Transactions on Sustainable Energy* 8.2 (2017), pp. 658–670.

[12] S. Sebastio, G. Gnecco, and A. Bemporad. "Optimal distributed task scheduling in volunteer clouds". *Computers & Operation Research* 81.C (May 2017), pp. 231–246.

[13] Y. Kuwata, A. Richards, T. Schouwenaars, and J. P. How. "Distributed robust receding horizon control for multivehicle guidance". *IEEE Transactions on Control Systems Technology* 15.4 (2007), pp. 627–641.

[14] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. "Distributed optimization and statistical learning via the alternating direction method of multipliers". *Foundations and Trends in Machine Learning* 3.1 (2011), pp. 1–122.

[15] A. Bemporad and M. Morari. "Control of systems integrating logic, dynamics, and constraints". *Automatica* 35.3 (1999), pp. 407–427.

[16] S. Li, R. R. Negenborn, and G. Lodewijks. "Distributed constraint optimization for addressing vessel rotation planning problems". *Engineering Applications of Artificial Intelligence* 48 (2016), pp. 159–172.

[17] B. Yang and M. Johansson. "Distributed optimization and games: a tutorial overview". *Networked Control Systems*. London: Springer London, 2010, pp. 109–148.

[18] R.R. Negenborn, B. De Schutter, and J. Hellendoorn. "Multi-agent model predictive control for transportation networks: Serial versus parallel schemes". *Engineering Applications of Artificial Intelligence* 21.3 (2008), pp. 353–366.

[19] L. Li, R. R. Negenborn, and B. De Schutter. "Distributed model predictive control for cooperative synchromodal freight transport". *Transportation Research Part E: Logistics and Transportation Review* (2016).

[20] Y. Kuwata and J. P. How. "Cooperative distributed robust trajectory optimization using receding horizon MILP". *IEEE Transactions on Control Systems Technology* 19.2 (2011), pp. 423–431.

[21] I. A. Khouy, P. Larsson-Kråik, A. Nissen, and U. Kumar. "Cost-effective track geometry maintenance limits". *Proceedings of the Institution of Mechanical Engineers, Part F: Journal of Rail and Rapid Transit* 230.2 (2016), pp. 611–622.