Train design and routing optimization for evaluating criticality of freight railroad infrastructures

Abdullah A. Khaled a, Mingzhou Jin b,*, David B. Clarke c, Mohammad A. Hoque d

a Mississippi State University, United States
b Department of Industrial and Systems Engineering, The University of Tennessee, Knoxville, 525D John D. Tickle Engineering Building, 851 Neyland Drive, Knoxville, TN 37996, United States
c The University of Tennessee, Knoxville, United States
d Texas Southern University, United States

Abstract
Freight transportation by railroads is an integral part of the U.S. economy. Identifying critical rail infrastructures can help stakeholders prioritize protection initiatives or add necessary redundancy to maximize rail network resiliency. The criticality of an infrastructure element, link or yard, is based on the increased cost (delay) incurred when that element is disrupted. An event of disruption can cause heavy congestion so that the capacity at links and yards should be considered when freight is re-routed. This paper proposes an optimization model for making-up and routing of trains in a disruptive situation to minimize the system-wide total cost, including classification time at yards and travel time along links. Train design optimization seeks to determine the optimal number of trains, their routes, and associated blocks, subject to various capacity and operational constraints at rail links and yards. An iterative heuristic algorithm is proposed to attack the computational burden for real-world networks. The solution algorithm considers the impact of volume on travel time in a congested or near-congested network. The proposed heuristics provide quality solutions with high speed, demonstrated by numerical experiments for small instances. A case study is conducted for the network of a major U.S. Class-I railroad based on publicly available data. The paper provides maps showing the criticality of infrastructure in the study area from the viewpoint of strategic planning.

1. Introduction and literature review
Railroads play a significant role in U.S. freight transportation to support the nation’s supply chain and national security (Davis et al., 2012; Stodolsky, 2011). Total U.S. rail freight ton-miles have doubled and density (measured by total ton-miles per mile of track) tripled between 1980 and 2006 (Cambridge Systematics, 2007). During the same period, total ton-miles carried by Class-I railroads increased by 93 percent (Eakin et al., 2009). In contrast, freight rail infrastructure has shrunk as carriers sought to reduce costs. Rail miles have decreased by 42 percent since 1980. In consequence, the U.S. freight railroad industry is currently operating without much excess capacity. A study by the American Association of State Highway and Transportation Officials (AASHTO) shows that projected rail freight movement will increase by 84 percent by 2035 (AASHTO, 2007). To promote railway freight security and resilience, the Transportation Security Administration (TSA) has...
established a freight rail division with the vision of “ensuring the secure movement of all cargo on our nation’s freight rail systems and promoting the free flow of commerce by working with our public and private sector partners to maintain a secure, resilient, and sustainable network” (TSA, 2012). To support this mission, rail infrastructure of high consequence and vulnerability need to be identified so that rail stakeholders can take effective measures to protect critical infrastructure.

This paper proposes a solution approach to determine the criticality of railroad network infrastructure at the strategic level. The criticality of an infrastructure element, a node or a link, is evaluated by estimating the increased cost (delay) when this element is disrupted. After a disruption, traffic needs to be re-routed over the residual network to keep freight flow. The disruption may cause huge congestions, directly or indirectly, over a big part of the network and make the flows at many links and yards close to their capacities. A system-wide optimization model and a solution approach considering the congestion effects and capacity restrictions are therefore necessary to route traffic and further evaluate the effect of a disruption.

Railroad operations are highly complex so that real-world problems are large in size (Assad, 1981). Early optimization attempts had limited success and could not incorporate many of real-life characteristics (Assad, 1980b; Haghani, 1987; Cordeau et al., 1998; Newman et al., 2002). In recent years, railroads have started to implement optimization-based decision support systems to address various operational issues, such as blocking planning, Block-to-Train Assignment (BTA), train scheduling, locomotive scheduling, crew scheduling, empty car movement, and so on. Modern optimization models and solution approaches are founded on more sophisticated problem formulations, the ever-increasing computing capacity, and much improved data (Brännlund et al., 1998; Barnhart et al., 2000; Ahuja et al., 2007; Jha et al., 2008). Freight rail operations often start by developing a blocking plan. It aggregates a vast number of shipments into blocks as they move together to reduce reclassification at yards and overall intermediate handling costs (Newton et al., 1998; Barnhart et al., 2000; Jha et al., 2008; Yue et al., 2011). Based upon a blocking plan, train routing (TR) or scheduling identifies train routes, timetables, and frequencies (Farvolden and Powell, 1994; Brännlund et al., 1998). BTA is then conducted to determine which trains should carry which blocks (Kwon et al., 1998; Jha et al., 2008). The combination of the two tasks of BTA and TR is called train design, which is a highly combinatorial and complex optimization problem. A number of efforts have been made to solve this problem using different cost terms and business constraints. Most of these researches divided the overall problem into two sub-problems of train design and block routing and solved them iteratively (Assad, 1980a; Haghani, 1987, 1989; Keaton, 1989, 1992; Marin and Salmerón, 1996). Gorman (1998) and Newman and Yano (2001) used the integrated approach to solve the whole problem. More recently, a few works addressed the large-scale train design problems (Ahuja et al., 2005; Lozano et al., 2011; Colombo et al., 2011; Wang et al., 2011; Jin et al., 2013). However, all above models or solutions approaches in literature studied train design under normal conditions without considering the capacities at yards and links and the congestion factor that may occur during any catastrophes. Cacciatori et al. (2014) recently provided an overview of real-time models and algorithms for recovery from disruptions or disturbances. In practice, North American railroads widely use the MultiModal system to create the sequence of blocks and train routes with both optimization and simulation techniques (Ireland et al., 2004). MultiModal creates block sequences by minimizing the number of switches and total car-miles under various constraints through shortest-path algorithms. The system is used for railway planning under normal conditions without considering congestion. In addition to freight train design, a lot of studies have been done for passenger train line planning and routing (e.g., Carey, 1994; Kaspi and Raviv, 2013), which do not have the classification issue but focus more on timetables. Caprara et al. (2011) provided an overview of optimization problems in passenger train planning.

Our research focuses on the train design problem in re-routing freight traffic to determine rail infrastructure criticality for the strategic planning purpose. The re-routing under disruptions is different from normal train design that disruptions could cause serious congestations at railroad links and nodes. This paper develops a train design approach that incorporates capacity constraints at links and yards, operational constraints of trains, and the volume-speed relationships at links. The criticality of one link or node is evaluated by the increased transportation cost (delay) caused by its disruption to the baseline case. Each disruption scenario is solved by the proposed train design approach.

The remainder of the paper is organized as follows. Section 2 presents an optimization model for train design and routing under congestion. Section 3 describes a solution approach based on a decomposition of the overall problem into sub-problems of BTA and TR and shows its performance with a small instance. A case study is performed for a real-world case on a major Class-I railroad in Section 4. Section 5 presents conclusion and discusses possible model extensions.

2. Mathematical formulation considering congestions

Different from train forming and routing under normal conditions, a disruptive event can cause a rail network to operate under high congestion and close to its capacity at both links and yards. An optimization model is necessary to incorporate the following congestion concerns. In the literature, the train design optimization models for a rail network under normal conditions do not consider the capacity at yards and links or speed-volume relationship at links (e.g., Colombo et al., 2011; Jin et al., 2013).

At each link, the travel time depends on the number of trains traveling on the link. When more trains are assigned to it, the speed on the link may be reduced. The decreasing speed-volume relationship depends on link features, such as the number of tracks, siding, and its signal system. Furthermore, the model also has an upper bound on the daily number of trains that can travel on each link. A recent research by Norfolk Southern (NS) Railroad (Yoon et al., 2011) tried to use a numerical
way to calculate this daily upper bound on their rail links. Shih et al. (2014) studied capacity expansion on links by sparse sidings.

At each classification yard, increased volume may increase the waiting time for an inbound train to be humped but may decrease the waiting time to form an departing train so that the cost (or time) for a railcar to be classified at one specific yard is assumed to be independent from the volume. However, a yard can only handle a limited number of trains and/or cars. If the volume exceeds the capacity, the traffic may overflow and block the traffic on the main tracks. This assumption is consistent to the numerical results presented by Petersen (1977) and Turnquist and Daskin (1982) and simulation results by Marinov and Viegas (2009). Their results showed that the dwell time at a yard keeps almost the same until the volume reaches a threshold value (capacity). Beyond that value, the dwell time goes up rapidly. The capacity concern is not important under the current normal practice because freight trains in the U.S. usually follow a timetable, which does not change daily. Furthermore, most yards are currently operated within their capacities.\footnote{This yard capacity assumption was made based on authors' interactions with CSX and NS railroads, two U.S. Class-I railroads.}

The following train forming and routing model with congestion concerns considers a directed railway network $G = (N, A)$, where $N$ is the set of nodes and $A = \{(i,j):i,j \in N\}$ is the set of directed links. $(i,j) \in A$ implies $(j,i) \in A$. $N_c$ is the set of nodes for classification and $N_c \subseteq N$. At each classification node (also known as yard) $i \in N_c$, a capacity is defined by the number of trains and the number of cars that the yard can handle every day, $U^c_i$ and $U^t_i$ respectively. The cost (or dwell time) for a railcar at yard $i \in N_c$ is assumed at $C^c_i$, which is independent from the classification volume. The dwell time in a yard depends on the layout and connection plan of the yard, as shown by Petersen (1977). In practice, railroads have historical data and also often use simulation to estimate the capacity for a specific yard. The distance of each link $(i,j) \in A$ is $d_{ij}$. The travel cost (related to travel time) per railcar depends on the daily number of trains traveling on the link. If $l$ trains travel on link $(i,j)$ per day, including both directions, the travel cost per railcar is $C^{t}_{ij}$ because more assigned trains usually mean lower speeds (Lai and Barkan, 2009), especially for single tracks. At most $U^M_{ij}$ trains, on both directions, can be assigned to link $(i,j)$, where $i < j$. Furthermore, the train length and weight are limited by $U^L_{ij}$ and $U^W_{ij}$ on each link $(i,j) \in A$, respectively, which are usually decided by link features. In addition, each train incurs a start cost of $C^{s}_{ij}$ and travel cost $C^{t}_{ij}$ for passing link $(i,j)$. $C^{s}_{ij}$ is the product of the train travel cost $C^{t}_{ij}$ per mile and the link distance $d_{ij}$. Both $C^{s}_{ij}$ and $C^{t}_{ij}$ are independent from the travel time and are therefore independent from the volume over links. All the four cost items ($C^{s}_{ij}, C^{t}_{ij}, C^{L}_{ij}, C^{W}_{ij}$) are normalized into a momentary unit though several of them are originally based on ton-hours.

A set of blocks $B$ need to be shipped from their origins to destinations, where $b$ is its index. For each block $b$, we know the following information: number of railcars $\gamma_b$, origin and destination $o_b$ and $d_b$, length $L_b$, and weight $W_b$. Whenever a train stops to either pick up or drop off blocks, it is called a work event. Each train cannot exceed its maximum allowable intermediate work events $R_{\text{max}}$ in its route, excluding the train origin and destination. Each train is restricted to carry a maximum number of blocks $R^M$ along its whole route. Without the loss of generality, this paper further makes the following assumptions.

- Each train can depart from a node at most once;
- A train can start and end at the same node; and
- A train can only pick up and drop off one block by at most once.

A mixed integer program (1)–(29) is developed to represent the described problem with the following decision variable definitions.

\[
\begin{align*}
 z^t_{ij} & : \text{whether train } t \text{ travels along link } (i,j) \in A; \\
y^l_{ij} & : \text{whether there are } l \text{ trains traveling along link } (i,j) \text{ on both directions, } i < j, l \in \{0, \ldots, U^M_{ij}\}; \\
x^t_i & : \text{whether train } t \text{ has a working event at node } i \in N; \\
u^s_i \text{ (} u^t_i \text{)} & : \text{whether train } t \text{ starts (ends) at node } i \in N; \\
a^t_i & : \text{whether train } t \text{ starts and ends at node } i \in N; \\
p^{s,t}_{ij} \text{ (} q^{s,t}_{ij} \text{)} & : \text{whether train } t \text{ picks up (drops) block } b \text{ at node } i \in N; \\
g^t_i \text{ (} h^t_i \text{)} & : \text{length (tonnage) of train } t \text{ after visiting node } i \in N; \\
o^t_i & : \text{number of railcars carried by train } t \text{ after passing node } i \in N; \\
\gamma^t_i & : \text{number of railcars carried by train } t \text{ after being classified at yard } i \in N_c; \\
\varphi^t_i & : \text{number of railcars that train } t \text{ carries along link } (i,j) \in A; \\
\tau_{ij} & : \text{total car travel cost over link } (i,j) \in A, i < j, \text{ for both directions;} \text{ and} \\
k^t_i & : \text{an artificial variable for eliminating sub-tours, } i \in N, t \in T.
\end{align*}
\]
\[
\begin{align*}
\text{Min} & \quad \sum_{t \in T} \sum_{(i,j) \in A} C_{ij}^t x_{ij}^t + C_{1}^t \sum_{t \in T} u_1^t + \sum_{i \in N, t \in T} C_{1}^t y_i^t + \sum_{t \in T} \tau_t \\
\text{S.T.} & \quad \sum_{(i,j) \in A} x_{ij}^t - \sum_{(j,i) \in A} x_{ji}^t = u_t^t - u_{t-1}^t \quad \forall t \in T, \forall i \in N \tag{1} \\
& \quad \sum_{(i,j) \in A} x_{ij}^t - \sum_{(j,i) \in A} x_{ji}^t = u_t^t - u_{t-1}^t \quad \forall t \in T, \forall i \in N \tag{2} \\
& \quad \sum_{(i,j) \in A} x_{ij}^t - \sum_{(j,i) \in A} x_{ji}^t = 0 \quad \forall t \in T, \forall i \in N \tag{3} \\
& \quad \sum_{t \in T} \sum_{i \in \{0, \ldots, \ell_{ij}^t\}} y_{ij}^t = 1 \quad \forall (i,j) \in A, i < j \tag{4} \\
& \quad \sum_{t \in T} \sum_{i \in \{0, \ldots, \ell_{ij}^t\}} y_{ij}^t = 1 \quad \forall (i,j) \in A, i < j \tag{5} \\
& \quad \sum_{t \in T} u_1^t \leq 1 \quad \forall t \in T \tag{6} \\
& \quad \sum_{t \in T} u_t^t \leq 1 \quad \forall t \in T \tag{7} \\
& \quad 2u_t^t \leq u_{t-1}^t + u_t^t \quad \forall t \in T, \forall i \in N \tag{8} \\
& \quad k_i^t \geq k_i^t + 1 - M d_i^t - \left(1 - z_i^t \right) \quad \forall t \in T, \forall (i,j) \in A \tag{9} \\
& \quad \begin{cases} 
\sum_{t \in T} p_{ij}^t = 1 - \sum_{t \in T} q_{ij}^t = 1 & \text{if } i = o \\\n\sum_{t \in T} q_{ij}^t = 1 - \sum_{t \in T} p_{ij}^t = 1 & \text{if } i = d \\\n\sum_{t \in T} p_{ij}^t = \sum_{t \in T} q_{ij}^t & \text{if } i \in N_o - o - d \end{cases} \quad \forall b \in B, \forall i \in N \tag{10} \\
& \quad u_t^t + x_i^t \geq p_{ij}^t \quad \forall t \in T, \forall i \in N_o, \forall b \in B \tag{11} \\
& \quad u_t^t + x_i^t \geq q_{ij}^t \quad \forall t \in T, \forall i \in N_o, \forall b \in B \tag{12} \\
& \quad u_t^t + x_i^t \geq \sum_{(i,j) \in A} z_{ij}^t \quad \forall t \in T, \forall i \in N \tag{13} \\
& \quad \sum_{t \in T} p_{ij}^t = \sum_{t \in T} q_{ij}^t \quad \forall b \in B, \forall t \in T \tag{14} \\
& \quad k_i^t \geq k_i^t + 1 + M \left( p_{ij}^t + q_{ij}^t - u_t^t - 2 \right) \quad \forall t \in T, \forall i, j \in N, \forall b \in B \tag{15} \\
& \quad g_i^t \geq g_i^t \quad \forall t \in T, \forall (i,j) \in A \tag{16} \\
& \quad (u_t^t - 1)M + \sum_{b \in B} p_{ij}^t L_b - \left(1 - z_i^t \right) \quad \forall t \in T, \forall (i,j) \in A \tag{17} \\
& \quad \sum_{b \in B} p_{ij}^t L_b \leq \sum_{b \in B} U_{ij}^t \quad \forall t \in T, \forall i \in N \tag{18} \\
& \quad h_i^t \geq h_i^t + \sum_{b \in B} p_{ij}^t w_b - \left(1 - z_i^t \right) \quad \forall t \in T, \forall (i,j) \in A \tag{19} \\
& \quad (u_t^t - 1)M + \sum_{b \in B} p_{ij}^t w_b \preceq \sum_{b \in B} U_{ij}^t \quad \forall t \in T, \forall i \in N \tag{20} \\
& \quad \phi_{ij}^t \geq \phi_{ij}^t - M \left(1 - z_i^t \right) \quad \forall t \in T, \forall (i,j) \in A \tag{22} \\
& \quad \sum_{t \in T} x_i^t \leq u_1^t \quad \forall i \in N \tag{23} \\
& \quad y_i^t \preceq y_i^t - M \left(1 - x_i^t - u_1^t \right) \quad \forall t \in T, \forall i \in N \tag{24} \\
& \quad \sum_{t \in T} y_i^t \preceq U^t \quad \forall i \in N \tag{25} \\
& \quad \sum_{t \in T} x_i^t \preceq \sum_{t \in T} y_i^t \quad \forall t \in T \tag{26} \\
& \quad \sum_{b \in B} p_{ij}^t \preceq \sum_{t \in T} y_i^t \quad \forall t \in T \tag{27} \\
& \quad \tau_t \geq \min \left\{ \sum_{t \in T} \left( \phi_{ij}^t + \phi_{ij}^t \right) - M \left(1 - y_i^t \right) \quad \forall (i,j) \in A, i < j, l \in \{0, \ldots, U_{ij}^t\} \right\} \tag{28} \\
& \quad \sum_{t \in T} u_1^t \leq \sum_{t \in T} u_t^t \quad t = 1, \ldots, T - 1 \tag{29} \\
& \quad x_i^t, z_i^t, u_t^t, p_{ij}^t, q_{ij}^t, y_i^t \in \{0, 1\}; g_i^t, h_i^t, \phi_{ij}^t, \phi_{ij}^t, \tau_t \geq 0 
\end{align*}
\]
The objective function (1) minimizes the total cost, including train travel cost, train start cost, railcar classification costs, and railcar travel cost along links, which depends on traffic volume. Constraint set (2) keeps the train flow conservation at all nodes. Constraint sets (3) and (4) make sure that one train can leave each node (link) at most once. Constraint sets (5) and (6) are used to obtain the number of trains traveling over link (i,j), represented by \( y^j_t \). They also make sure that the number of trains on both directions over link (i,j) will not exceed its capacity of \( U^j_{i} \). Constraint set (7) guarantees that each used train only starts once while constraint set (8) permits a train to start and end at the same node. Sub-tours are eliminated by constraint set (9). When node \( j \) is visited by train \( t \) immediately after node \( i \), \( k^j_t \) will be greater than \( k^i_t \) by at least 1. Here, \( M \) is a big number. Constraint set (10) guarantees a block is picked up (dropped off) by a train at either its origin (destination) or in the classification yard where some train drops (picks) it and a block is not dropped (picked) at its origin (destination). Constraint sets (11) and (12) together make \( y^i_t = 1 \) when train \( t \) picks or drops (except its terminating yard) some blocks at yard \( i \). Constraint set (13) guarantees that a train definitely passes the node where the train has a work event (classification). When a train picks up a block, the train must drop it, as indicated by (14). A train must pick up a block before it can drop that block, guaranteed by constraint set (15) using the ordering variable of \( k^i_t \) obtained from constraint set (9). Please note that (15) does not matter when the dropping point is the termination node of the train, which is necessary when a train starts and terminates at the same node. The length of train \( t \) after visiting a node is obtained by constraint sets (16) and (17) and satisfies the maximum train length restriction on each link. Similarly, the weight restriction of trains over each link is met through constraint sets (18) and (19). The number of railcars a train has after visiting a node is obtained by constraint sets (20) and (21). The number of railcars that a train carries over a link is obtained by constraint set (22). Constraint set (23) restricts the number of trains that can be classified at each yard \( j \) while constraint sets (24) and (25) restrict the number of railcars each classification yard can handle. Constraint set (26) restricts the maximum number of intermediate work events a train can have. The number of blocks one train can carry cannot exceed the maximum number of blocks that a train is allowed to transport, restricted by constraint set (27). Constraint set (28) determines the railcar travel cost traveling both directions over each link, which is used in the objective to calculate the total railcar travel cost for all links. Finally, Constraint set (29) is used to reduce the symmetry of solutions to improve the computational efficiency.

In summary, the model of (1)–(29) has three major parts. Constraint sets (2)–(9) are for train routing. Constraint sets (10)–(15) are used to make sure all blocks are served well by trains. Constraint sets (16)–(27) are used to guarantee that various capacity restrictions are met, including yard capacity, link capacity, and the restriction of train length and weight on each link. This MIP model is a highly combinatorial optimization problem that cannot be solved for a large network using exact optimization techniques, though small instances can be solved optimally using commercial software (e.g., IBM CPLEX), as shown in Table 1 in Subsection 3.4. Hence, a heuristic algorithm is proposed in the next section to obtain quality solutions within a reasonable amount of computational time.

### 3. Solution approach

An iterative heuristic algorithm, illustrated by Fig. 1, is developed with two major sub-problems of BTA and TR. The inputs include network data, block data, and all the cost parameters and restrictions. The initialization stage creates a sub-network for each block and assigns the blocks with reasonably large number of railcars to ‘unit’ trains, which travel from their origins to destinations without any classifications. The BTA algorithm assigns the remaining blocks one by one to trains and provides a list of trains that are formed to serve all the blocks. At the TR stage, formed trains are routed optimally with their known stops at classification yards. BTA and TR stages are repeated until the percent cost gap between two consecutive TR runs falls below a predefined tolerance limit \( \Delta \).

### Table 1

Results of the optimal and heuristic solution approaches for small instances.

<table>
<thead>
<tr>
<th>Instance</th>
<th>Number of Nodes-Arcs-Blocks</th>
<th>CPLEX Value ($)</th>
<th>CPLEX CPU Time (sec.)</th>
<th>Heuristic Solution Value ($)</th>
<th>Heuristic Solution CPU Time (sec.)</th>
<th>Difference Value</th>
<th>Difference CPU Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>4-5-4</td>
<td>15387.5</td>
<td>131.8</td>
<td>15387.5</td>
<td>0.11</td>
<td>0.000%</td>
<td>99.947%</td>
</tr>
<tr>
<td>2</td>
<td>5-7-5</td>
<td>17377.5</td>
<td>921.6</td>
<td>17377.5</td>
<td>0.19</td>
<td>0.000%</td>
<td>99.983%</td>
</tr>
<tr>
<td>3</td>
<td>6-9-6</td>
<td>23637.5</td>
<td>1838.2</td>
<td>23637.5</td>
<td>0.28</td>
<td>0.000%</td>
<td>99.987%</td>
</tr>
<tr>
<td>4</td>
<td>7-11-7</td>
<td>29328.8</td>
<td>4234.5</td>
<td>29328.8</td>
<td>0.39</td>
<td>0.000%</td>
<td>99.989%</td>
</tr>
<tr>
<td>5</td>
<td>8-13-8</td>
<td>40112.5</td>
<td>6598.9</td>
<td>40112.5</td>
<td>0.46</td>
<td>0.000%</td>
<td>99.992%</td>
</tr>
<tr>
<td>6</td>
<td>9-15-9</td>
<td>44098.2</td>
<td>9989.1</td>
<td>44098.2</td>
<td>0.59</td>
<td>0.000%</td>
<td>99.993%</td>
</tr>
<tr>
<td>7</td>
<td>10-17-10</td>
<td>47216.6</td>
<td>12874.0</td>
<td>47216.6</td>
<td>0.73</td>
<td>0.000%</td>
<td>99.994%</td>
</tr>
<tr>
<td>8</td>
<td>11-19-11</td>
<td>51462.3</td>
<td>18985.4</td>
<td>51462.3</td>
<td>0.91</td>
<td>0.000%</td>
<td>99.995%</td>
</tr>
<tr>
<td>9</td>
<td>12-21-12</td>
<td>54539.6*</td>
<td>20000.0</td>
<td>54445.8</td>
<td>1.01</td>
<td>0.172%</td>
<td>99.994%</td>
</tr>
<tr>
<td>10</td>
<td>13-23-13</td>
<td>56768.8*</td>
<td>20000.0</td>
<td>56135.5</td>
<td>1.28</td>
<td>1.115%</td>
<td>99.993%</td>
</tr>
</tbody>
</table>

\* Best solution when the time limit of 20,000 s is reached.
3.1. Initialization Step

In the Initialization step, detailed in Fig. 2, a sub-network for each individual block from the whole network is built to reduce computational burden. Rather than the whole network \((N,A)\), only a sub-network of \((N_b,A_b)\) in which all nodes have a distance of less than \(D\) miles from the shortest path for the block are considered when routing block \(b\). Fig. 3 shows an example of a 10-node network with links and their distances. If we consider a block \(b\) whose \(o_b = 2\) to \(d_b = 4\) (i.e. the block is from yard 2 to yard 4), the shortest path for block \(b\) is obviously \(P_b = 2 → 1 → 4\) with the distance of 80. If we assume

*Initialization*

Create a shortest path for each pair of two nodes \(i,j \in N\) and have the shortest distance \(D_{ij}\).

For each block \(b\) in \(B\):

- Include all nodes in the shortest path \(P_b\) from \(o_b\) to \(d_b\) into set \(N_b\);
- Add all nodes in \(N \setminus N_b\) whose distance from any node in \(P_b\) is less than or equal to a given real number \(D\) into \(N_b\); and
- Let \(A_b\) the set of links with both ends in \(N_b\).

For each block \(b\) in \(B\) with \(r_b ≥ R\) (in the case study, \(R = 80\) railcars):

- Assign ‘unit’ trains for \(b\), each following the shortest path \(P_b\) that satisfies link weight and length restriction along the path;
- Update the available train capacity \(U^{mf}_{ij}\) at links \((i,j) \in P_b\) and \(U_f^c\) at the origin; and
- Update \(r_b, l_b\), and \(w_b\) by deducting railcars carried by ‘unit’ trains.

**Fig. 2.** Steps of initialization algorithm for creating block sub-networks and ‘Unit’ trains.

---

**Fig. 1.** Framework of the proposed heuristic algorithm.
\( D = 40 \), we have \( N_b = \{1,2,4,5,7,9\} \) and \( A_b = \{(1,2), (2,1), (1,4), (4,1), (1,9), (9,1), (2,7), (7,2), (4,5), (5,4), (4,9), (9,4)\} \). The new reduced network \((N_b,A_b)\) will be used later for routing block \( b \). If the demand from node 2 to node 4 is 100 railcars, a unit train with \( R = 80 \) railcars is travelling directly from node 2 to node 4 without classification when the train does not violate the weight and length restrictions on links (2,1) and (1,4). The remaining 20 railcars will be considered as the demand for this OD pair. At the same time, the capacity at involved yards and links (i.e., (1,2) and (1,4)) are reduced.

### 3.2. Block-to-Train Assignment (BTA)

A set of trains \( T \) serving blocks \( b \in B \) is assumed to be obtained from the previous iteration. The given information at the beginning of BTA is as follows.

For each train \( t \in T \):
- The route from node \( o_t \) to node \( d_t \), the set of visited nodes \( N_t \), and the set of visited links \( A_t \),
- The number of carried railcars \( Q_{it} \), length \( g_{it} \), and weight \( h_{it} \) after visiting node \( i \in N_t \), and
- The number of work events \( X_t \), the set of carried blocks \( B_t \), and their pickup and drop-off points.

For each classification yard \( i \in N_c \):
- The number of classified trains, \( \lambda_i \), and
- The number of classified railcars, \( \gamma_i \).

For each link \((i,j) \in A\):
- The total number of railcars carried by all trains, \( \phi_{ij} \), and
- The total number of traveled trains, \( l_{ij} \).

BTA, explained in Fig. 4, reassigns blocks in \( B \), one by one following a random order, to existing trains or a new train with the least cost. The random order in Step 1 is used to avoid local optimal solutions over iterations. Train information is updated along with the network capacity consumptions after assigning a block. At Step 3, the set of candidate trains \( T_b \) that can carry block \( b \in B \) is formed by deleting the trains that (1) have carried \( R_c \) blocks belonging to \( B -\{b\} \), (2) have carried only block \( b \), (3) are unit trains, or satisfy (4) \( A_t \cap A_b = \emptyset \) (called the train selection criteria in Fig. 4). Here, \( A_b \) is the candidate links for block \( b \) and obtained in the Initialization step.

A new variable of \( \rho_{ij}^t \) is introduced to indicate whether train \( t \) is classified at node \( i \in N_t \) in Step 4.

\[ \rho_{ij}^t = \begin{cases} 0 & \text{if train } t \text{ picks up/drops any blocks in } B - \{b\} \text{ at node } i \in N_t \\ 1 & \text{otherwise.} \end{cases} \]

At step 4, a new directed network for each train \( t \in T_b \) is built for block \( b \) that is under consideration. If \( o_b \in N_t \) and/or \( d_b \in N_t \), we delete all nodes visited before \( o_b \) and all nodes visited after \( d_b \) from \( N_t \) to have \( N'_t \), and delete all links visited before \( o_t \) or after \( d_t \) from \( A_t \) to have \( A'_t \). If \( A'_t = \emptyset \), we don’t build a network and skip all future steps for the train. We then delete any links that will exceed its length and weight maximum if train \( t \) carries block \( b \) from \( A'_t \). For each pair of nodes \( m, n \in N'_t \), when there is a path from \( m \) to \( n \) in \((N'_t,A'_t)\) and \( \rho_{im}^t + \rho_{in}^t + X_t \leq R_c \), \( \lambda_m + \rho_{im}^t \leq U_{im}^L \), \( \gamma_m + \rho_{im}^t \leq U_{im}^C \), \( \gamma_n + \rho_{in}^t \leq U_{in}^C \), a new link directly from \( m \) to \( n \) is added into \( A'_t \) with the cost \( CN_{im}^{lt} \) based on the shortest path from \( m \) to \( n \) in the network of

**Fig. 4.** Block-to-Train Assignment (BTA) Algorithm.
(N\textsubscript{b}, A\textsubscript{b}'). Adding this link indicates that train \( t \) to carry block \( b \) from \( m \) to \( n \) without violating any constraints. The total cost of using train \( t \) to carry block \( b \) from \( m \) to \( n \) is denoted by \( C^t_{mn} \) and can be calculated by (30).

\[
C^t_{mn} = D^t_{mn}r_b + (\rho^t_{m} + r_b)C^C_m + \rho^t_{n}C^C_n + \sum_{k \in \mathcal{P} \cup \{m \} \cup \{n \}} (1 - \rho^t_{k})r_b C^C_k. \tag{30}
\]

The shortest path cost \( C^t_{mn} \) is calculated by running the shortest cost algorithm in \((N_b, A_b')\) and the cost to travel on each link \((i, j) \in A_b'\) is \( C^C_{ij} \). Here, \( C^C_{ij} \) is the travel cost of each raillcar on link \((i, j) \in A_b'\) when \( b \) trains have already traveled on that link.

\[
(p^t_m + r_b)C^C_m + \rho^t_n C^C_n + \sum_{k \in \mathcal{P} \cup \{m \} \cup \{n \}} (1 - \rho^t_k) r_b C^C_k \text{ is the additional classification cost along the train path, } P_{m \rightarrow n}, \text{ for picking up block } b \text{ at classification yard } m \text{ and dropping the block at yard } n.
\]

If \( o_b(\text{or } d_b) \text{ is NOT in } N_b, \text{ then } d' \neq o', \text{ and } \sum_{i \in N} (1 - \rho^t_i) + 1 \leq R^{a\text{w}}, \text{ we extend the network of } (N_b, A_b') \text{ to } o_b(\text{or } d_b). \text{ We add } o_b \text{ into } N_b \text{ with a link from } o_b \text{ to each node } n \in N_b \text{ into } A_b' \text{ if } 1 + \rho^t_n + x' \leq R^{a\text{w}}, \text{ and } \gamma_n + \rho^t_n d'_n \leq U^{a\text{w}}_n, \text{ with the cost of } C^t_{o_b n} \text{ as (31).}

\[
C^t_{o_b n} = D^t_{o_b n}r_b + TC^t_{d' o_b} + \rho^t_n d'_n C^C_n + C^C_o r_b + \sum_{k \in \mathcal{P} \cup \{o_b \} \cup \{n \}} (1 - \rho^t_k) r_b C^C_k. \tag{31}
\]

\(DC^t_{d' o_b}\) in (31) is the shortest path cost for train to taking the block from \( o' \) to \( n \) and is similar to \( DC^t_{mn}\) in (30). The shortest path cost \( TC^t_{d' o_b}\) over links to extend train \( t \)'s path so that it starts at \( o_b \) is calculated from \( o_b \) to \( o' \) and picks block \( b \) from \( o_b \) by

- **Updating the network of** \((N_b, A_b)\) **after removing the links with** \( U^t_m \) **trains and calculating the travel cost in each remaining link** \((i, j) \in A_b\) as

\[
C^C_{ij} = C^C_{ij} + \left( \rho^C_{ij} - C^C_{ij} \right) \phi_{ij} + C^T_{ij}, \text{ and}
\]

- **Running the shortest cost algorithm from** \( o_b \) **to** \( o' \) **in** \((N_b, A_b)\) **to obtain** \( TC^t_{d' o_b}\).

The remaining three terms are the additional classification costs for train \( t \) to take the block from \( o' \) to \( n \). By the above operations, \( o_b \) is added into \( N_b \).

Similarly, the destination of the block \( d_b \) is added into \( N_b \) if the following conditions are met.

1. \( d_b \neq N_b \) and
2. \( d_b \neq N_b \) and

We add \( d_b \) into \( N_b \) with a link from each node \( m \in N_b \) to \( d_b \) if \( 1 + \rho^t_m + x' \leq R^{a\text{w}}, \gamma_m + \rho^t_m d'_m \leq U^{a\text{w}}_m, \text{ and } \gamma_n + \rho^t_n d'_n \leq U^{a\text{w}}_n, \text{ with the cost of } C^t_{m d_b} \text{ as (33).}

\[
C^t_{m d_b} = D^t_{m d_b}r_b + TC^t_{d'_ m} + \left( \rho^t_m d'_m + r_b \right) C^C_m + \sum_{k \in \mathcal{P} \cup \{m \} \cup \{d_b \}} (1 - \rho^t_k) r_b C^C_k. \tag{33}
\]

\(DC^t_{d'_ m}\) is calculated similar to \( DC^t_{mn}\) in (30) and also \( TC^t_{d'_ m}\) is calculated similar to \( TC^t_{d' o_b}\).

If both \( o_b \) and \( d_b \) are newly added nodes and \( 2 + x' \leq R^{a\text{w}}, \gamma_n + 1 \leq U^{a\text{w}}_n, \lambda_d + 1 \leq U^{a\text{w}}_d, \text{ and } \gamma_n + \rho^t_n d'_n \leq U^{a\text{w}}_n, \text{ we add a direct link from } o_b \text{ to } d_b \text{ into } A_b' \text{ with the cost of } C^t_{o_b d_b}, \text{ shown in (34).}

\[
C^t_{o_b d_b} = \left( D^t_{o_b d' b} + D^t_{d'_ o b} + D^t_{d'_ d} + C^t_{d'_ b} \right) r_b + \sum_{k \in \mathcal{P} \cup \{o_b \} \cup \{d_b \}} (1 - \rho^t_k) r_b C^C_k. \tag{34}
\]

In Sep 5, a link from \( o_b \) to \( d_b \) is created as a new train with the cost of \( C^t_{mn} \) as (35) if \( \lambda_o + \rho^t_o \leq U^{a\text{w}}_o \) and \( \lambda_d + \rho^t_d \leq U^{a\text{w}}_d \) in \((N_b, A_b')\). Please note that the \( TC^t_{o_b d_b}\) is the minimum travel cost of routing a new train from \( o_b \) to \( d_b \), which includes the possibly increased cost for the existing trains. The concept is the same as the way to calculate \( TC^t_{d' o_b}\) by using (32).

\[
C^t_{o_b d_b} = \left( C^t_{o_b d'_ b} + C^t_{d'_ o b} + C^t_{d'_ d} \right) r_b. \tag{35}
\]

In Step 6 of BTA, all networks \((N_b', A_b')\) for \( t \in T^b \) are combined together into one network called \((N_b, A_b')\). For a link traveled by more than one train, the one with the least cost is selected. We create each link from \( o_b \) to \( n \in N_b \) as a new train and add it into \( A_b' \) with the cost of \( C^t_{mn} \) based on (36), if \( \lambda_o + \rho^t_o \leq U^{a\text{w}}_o \) and \( \lambda_n + \rho^t_n \leq U^{a\text{w}}_n \) and \( C^t_{mn} \) is less than the current cost of link \((o_b, n) \) in \((N_b, A_b')\).

\[
C^t_{mn} = D^t_{mn}r_b + C^t_{o_b d_b}. \tag{36}
\]
Similarly, we create each link from \( n \in N^C_b \) to \( d_b \) as a new train and add it into \( A^C_b \) with the cost of \( CN^\text{New}_{n,d_b} \) as (37), if \( \lambda_n + \rho^n_b \leq U^T_n \) and \( \lambda_{d_b} + \rho^T_{d_b} \leq U^T_{d_b} \) and \( CN^\text{New}_{n,d_b} \) is less than the current cost of link \((n,d_b) \) in \((N^C_b,A^C_b)\).

\[
CN^\text{New}_{n,d_b} = D^r_{n,d_b} + C^S + C^C_{n,d_b}.
\]

The next step (Step 7) in the BTA algorithm is to find the shortest cost path from \( o_b \) to \( d_b \) in the network of \((N^C_b,A^C_b)\). Each link in the shortest cost route represents a different train. Train information and network capacity information is updated after the assignment of a block. Please note that though the shortest-path algorithm is used frequently in BTA, the definition and the costs associated with those newly defined links in \( A^C_b \) are not the same as the physical links in the original network and the mileages of those physical links. Each newly-defined link \((i,j) \) in \((N^C_b,A^C_b)\) indicates that block \( b \) is classified at yard \( j \) right after yard \( i \). The congestion cost at links and classification cost at yards are considered in \( CN^t_{ij} \) if the block is carried by train \( t \) from yard \( i \) to yard \( j \). These calculations are very different from the shortest-path algorithms used by MultiModal (Ireland et al., 2004) on the original physical network.

### 3.3. Train Routing (TR)

Each train made up in BTA with known start, termination, intermediate work event nodes is re-routed through an optimization model at the TR step. Similar to block sub-networks, we also create a train sub-network for each train. Before routing a train, Pre-TR steps are used to remove routing information of the train under consideration so that congestion effects are not double counted.

**Pre-TR**

Each train \( t \in T \) carries a set of blocks \( b \in B^t \) with known pickup and drop-off points \((P^t_b, Q^t_b)\). The path of each train is known along with its \( X^t \) work event nodes and let \( N^t_p \) be the set of nodes in \( N^t \) with a working event, \( i.e. N^t_p = \bigcup_{b \in B^t} \{o^t_b, d^t_b\} \). A train sub-network is created to reduce problem size by

![Fig. 5. Quality of heuristic solution approach based on solution time.](image-url)

<table>
<thead>
<tr>
<th>Table 2</th>
<th>Sample data of class-I railroad network: link attributes.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Origin ID</td>
<td>Destination ID</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>5</td>
</tr>
<tr>
<td>3</td>
<td>5</td>
</tr>
<tr>
<td>3</td>
<td>6</td>
</tr>
</tbody>
</table>

* Capacity for each individual train traveling through the link.
Adding node $j$ into $N^t$ if there is a pair of nodes $(i,j)$ such that $i \in N^t, j \in N^t \setminus N^t$ and $D_{ij} \leq D$, where $D$ is a predefined number (e.g. 100 miles);

- Adding any links that have both ends in $N^t$ and are not currently in $A$ into $A$;

- Deleting the links in $A$ that have already reached the maximum train capacity restriction; and

- Letting $l_{ij}$ and $\varphi_{ij}$ be the numbers of trains and railcars, excluding train $t$, along link $(i,j) \in A^t$, where $l_{ij} = l_{ji}$ and $\varphi_{ij} = \varphi_{ji}$.

The TR optimization model is provided below with the following decision variable definition.

$z_{ij}$ whether the train travels along link $(i,j), (i,j) \in A$;

$u_i(t_i)$ whether the train starts (ends) at node $i, i \in N^t$;

$a_i$ whether the train starts and ends at node $i, i \in N^t$;

$g_i(h_i)$ length (tonnage) of the train after visiting node $i, i \in N^t$;

$\theta_i$ number of railcars carried by the train after visiting node $i, i \in N^t$;

$\tau_{ij}$ total additional car travel cost over link $(i,j) \in A$; and

$k_i$ an artificial variable to eliminate sub-tours, $i \in N^t$.

Minimize $\sum_{(i,j) \in A} c_{ij}z_{ij} + \sum_{(i,j) \in A} \tau_{ij}$

S.T.:

$\sum_{(i,j) \in A} z_{ij} - \sum_{(j,i) \in A} z_{ji} = u_i - v_i \quad i \in N^t$ (38)

$\sum_{(i,j) \in A} z_{ij} \leq 1 \quad i \in N^t$ (39)

$\sum_{(i,j) \in A} z_{ji} \leq 1 \quad i \in N^t$ (40)

$\sum_{i \in N^t} u_i = 1$ (41)

$2a_i \leq u_i + v_i \quad i \in N^t$ (42)

$k_{ij} \geq k_i + 1 - Ma_j - (1 - z_{ij})M \quad (i,j) \in A$ (43)

$\sum_{(i,j) \in A} z_{ij} + v_i \geq 1 \quad i \in N^t$ (44)

$g_i \geq g_i + \sum_{b \in B^t} p_{ib}L_{ib} - (1 - z_{ij})M - \sum_{b \in B^t} Q_{ib}L_{ib} - Mv_j \quad (i,j) \in A$ (45)

$(u_i - 1)M + \sum_{b \in B^t} p_{ib}L_{ib} \leq g_i + \sum_{b \in B^t} U_{ib} z_{ij} \quad i \in N^t$ (46)

$h_i \geq h_i + \sum_{b \in B^t} p_{ib}W_{ib} - (1 - z_{ij})M - \sum_{b \in B^t} Q_{ib}W_{ib} - Mv_j \quad (i,j) \in A$ (47)

$(u_i - 1)M + \sum_{b \in B^t} p_{ib}W_{ib} \leq h_i + \sum_{b \in B^t} U_{ib} z_{ij} \quad i \in N^t$ (48)

$\theta_{ij} \geq \theta_i + \sum_{b \in B^t} p_{ib}R_{ib} - (1 - z_{ij})M - \sum_{b \in B^t} Q_{ib}R_{ib} - Mv_j \quad (i,j) \in A$ (49)

$\theta_i \geq (u_i - 1)M + \sum_{b \in B^t} p_{ib}R_{ib} \quad i \in N^t$ (50)

$\tau_{ij} \geq \left(C_{ij}^{t+1} - C_{ij}^t\right)\varphi_{ij} + C_{ij}^{t+1}\theta_i - M(1 - z_{ij}) \quad (i,j) \in A$ (51)

$z_{ij}, u_i, v_i, \theta_i, \varphi_{ij}, k_i \geq 0$. 

### Table 3

Sample data of class-I railroad network: node attributes.

<table>
<thead>
<tr>
<th>Node ID</th>
<th>Capacity (cars)</th>
<th>Capacity (trains)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>2000</td>
<td>48</td>
</tr>
<tr>
<td>1</td>
<td>1600</td>
<td>40</td>
</tr>
<tr>
<td>2</td>
<td>2200</td>
<td>55</td>
</tr>
<tr>
<td>3</td>
<td>1500</td>
<td>35</td>
</tr>
<tr>
<td>4</td>
<td>1200</td>
<td>30</td>
</tr>
<tr>
<td>5</td>
<td>2600</td>
<td>65</td>
</tr>
<tr>
<td>6</td>
<td>1500</td>
<td>42</td>
</tr>
<tr>
<td>7</td>
<td>1200</td>
<td>32</td>
</tr>
<tr>
<td>8</td>
<td>1800</td>
<td>42</td>
</tr>
</tbody>
</table>
The objective function (38) minimizes the total cost, including train travel cost and railcar travel cost that depends on link volume, which is \( C_{ij}^{t+1} \). Constraint set (39) keeps the flow conversation. Constraint sets (40) and (41) make sure that train \( t \) can visit each node (link) at most once. Constraint set (42) ensures that the train can start only once while constraint set (43) permits the train to start and end at the same node. Sub-tours are eliminated by constraint set (44). Constraint set (45) guarantees that the train definitely leaves the node where the train has a work event, except the termination node. The train must pick up a block before dropping the block, guaranteed by constraint set (46) by using the variables ordering nodes visited by the train from (44). Please note that constraint set (46) is always loose when the dropping point is the termination node. Length restriction of the train over each link is ensured by constraint sets (47) and (48). Similarly, the weight restriction over each link is met by constraint sets (49) and (50). The number of railcars the train has after visiting a node is obtained by constraint sets (51) and (52). Finally, the additional railcar traveling cost caused by routing this train over a link is obtained by constraint set (53). Though the model (38)–(53) seems complicated, it is much smaller than the original model (1)–(29) because it has a much smaller network and is only for one train. Most of variables in the original model (1)–(29) become known (parameters) with values from BTA. The model for each train can be solved to its optimum with commercial solvers, such as IBM CPLEX. However, we note that the resulting solution may be suboptimal to the overall train routing problem.

### 3.4. Numerical experiments to show the computational effectiveness of the heuristic method

The iterative heuristic algorithm is implemented using C++ and calling IBM CPLEX 12.3 for the TR sub-problem on a PC with 3.30 GHz CPU and 16 GB RAM. The MIP model (1)–(29) of the overall problem is also solved to optimality for small instances by CPLEX 12.3 as a benchmark. Please note the optimization model (1)–(29) cannot be solved for large-scale instances. Table 1 summarizes the results for ten randomly generated small-sized instances. The second column reports the problem size in terms of the numbers of nodes, links, and blocks. The optimization model (1)–(29) could not be solved to its optimality by CPLEX within the 20,000-s time limit from Instance 9. The heuristic algorithm provides quality solutions with much higher speed (less than 1 s). For the first small instance, the heuristic algorithm reaches the same optimal solution from CPLEX. Table 1 justifies the superiority of the proposed heuristic algorithm regarding both solution quality and computational time. The computational time of the proposed heuristic algorithm is also shown in Fig. 5.

### 4. Case study

The proposed solution approach is applied to a Class-I railroad network to evaluate the criticality of its network infrastructures. The network consists of 200 nodes and 478 links. The case study evaluate the criticality of each element, a node or a link, by comparing the total travel cost (time) before and after the disruption to (removal of) this element. The study uses two types of data: network data and freight flow data. The flow data contain the 552 blocks’ ODs and their attributes (e.g., number of cars, lengths and tonnages). The network attributes include distance between stations/yards, number of tracks, and track operating characteristic (e.g., signal types). The base reference is the software North American Railroad Map, Version 3.11, which reflects the North American railroad network in 2010. Using the track attributes, we estimated the capacity in the number of trains per day based on Cambridge Systematics (2007) and other parameters (e.g., length and tonnage restrictions allowed for each traveled train) of each link. Information available at the railroad’s website aided this process.
Fig. 6. Criticality map of one Class-I railroad network.

Fig. 7. Criticality map in more details.
Station/Yard capacities (Marinov and Viegas, 2009) were derived from the railroad’s websites and correspondence with railroad personnel. Tables 2 and 3 provide sample network data for links and nodes, respectively.

Freight flow data were collected from the Freight Analysis Framework (FAF) database. It combines data from several sources to create a comprehensive image of freight movement among states and major zones by all transportation modes. FAF version 3 provides the most recent freight flow data of tonnage and value, by commodity type, mode and OD. This study uses 2009 rail freight data for all OD pairs by tonnage from (FHWA, 2009). Blocks per day for each OD are calculated based on (Cambridge Systematics, 2007). The sample freight flow data used in the case study is given in Table 4. Table 5 shows various cost parameters and train attributes collected from (Cambridge Systematics, 2007; Ahuja et al., 2005). Three levels of congestion: normal (no congestion), moderate, and high are considered at links.

The proposed heuristic solution approach, outlined in Fig. 1, is applied to the whole network to obtain the total cost, which is considered the ‘standard’ transportation cost. Subsequently, each network element, link or node, is considered to be disrupted individually (removed from the network) and all traffic is re-routed in the residual network with the heuristic approach. The criticality of each element is estimated by the increased transportation cost caused by the disruption to that element compared to the standard transportation cost. Fig. 6 demonstrates the criticality levels of all elements for this case, which is obtained by running the heuristic approach for about 900 times, one for the disruption of each element (link or node). Each run takes about 15 min. Fig. 7 shows the enlarged views of the criticality levels of the part of network infrastructures for better visibility and legibility. The relative criticality level of a link or yard is represented by its width or size. Wider links or bigger yards are more critical. A disruption to those elements will result in more transportation cost (delay) for the whole rail freight flows. The stakeholders of rail network should pay more attention to protect or add redundancy to those elements in order to enhance the whole network’s resilience.

5. Conclusion and future work

This paper evaluates the criticality of railroad network components by optimally creating and routing of trains through a disrupted and possibly congested network. Because of the expected congestion and capacity constraints, existing train forming and routing models and algorithms in the literature, which were developed for normal operations, could not be used under an event of disruption. This paper proposes an optimization model considering the capacity at both links and yards and the speed–volume relationship at links. An iterative heuristic solution method is proposed to solve the large-scale instances. The numerical study based on small instances shows that the heuristic solution approach can yield high quality solutions with high speed. The acceptance of computational speed is further verified by the case. Currently, we were told by several Class-I railroads that the freight rail network in US are in general running under its capacity. Both the model and the solution method proposed by this paper may not be appropriate for normal operational planning because the capacity and congestion concerns at links and nodes do significantly increase the complexity of the model. In fact, the normal train forming and routing may have to consider other factors, such as crew scheduling and routing, track rights, locomotive availability, timetables, etc. However, the methodology described in this paper may help them develop long-term strategic plans and develop preparedness and response plans for disruptions based on criticality level of infrastructure. Criticality of each individual link or yard is measured by the increased transportation cost (delay) after re-routing the trains through the residual network in the absence of the element. Another limit of this research is that the application of the proposed methodology in practice depends on capacity models on links and yards. Several on-going efforts to study railroad capacity supported by Transportation Research Board and the NuRail Center are expected to improve the applicability of this proposed method.

The method could also be used to estimate the effect of a certain disruption scenario that may involve multiple rail infrastructure elements. However, the paper does not demonstrate this application because of the large number of possible scenarios and therefore large computational burden. Our methodology for evaluating the criticality might be improved by developing a decision support system (DSS) based on this heuristic algorithm. It should have the feature to arbitrarily disrupt multiple network components and run the model to evaluate their criticality. The DSS may have more practical value if interfaced with the commercial geographic information system (GIS) software.

Acknowledgment

This research was partially supported by the US Department of Transportation, Office of the Secretary, Grant No. DTOS59-09-G-00058.

References


2 http://www.nurailcenter.org/