Supplementary Information for:

Topology shapes road network recovery: Global evidence from 224 cities

Srijith Balakrishnan and Patrick Stokkink

This Supplementary Information provides detailed mathematical formulations, simulation parameters, and statistical modeling procedures supporting the analyses reported in the main text. It is organized as follows:

Supplementary Note 1: Graph-theoretic formulation of road networks

Each road network is modeled as a directed graph $G = (V, A)$, where the set of vertices $V = \{v_1, v_2, \ldots, v_N\}$ represents road intersections and A denotes the directed arcs (road segments) connecting them. Two-way streets are represented as pairs of directed arcs.

We define $\alpha_{ij} = 1$ if two vertices v_i and v_j are directly connected by an arc, and 0 otherwise:

$$A = \{(v_i, v_j) \mid v_i, v_j \in V,\ \alpha_{ij} = 1\}.$$

Each arc (i, j) is assigned a travel time σ_{ij} proportional to link length, where $\sigma_{ij} = \infty$ if $\alpha_{ij} = 0$. The travel time between any pair of vertices v_r and v_s is the length of the shortest path σ_{rs}, and θ_{rs} denotes the number of such paths.

For any vertex or edge $x \in (V, A)$, $\theta_{rs}(x)$ represents the number of shortest paths between v_r and v_s that pass through x. This formulation enables the computation of topological metrics (degree, clustering, centrality, assortativity) that characterize network structure and underpin the restoration strategies.

Supplementary Note 2: Restoration strategies based on centrality measures

For each disrupted arc (i, j), the selected centrality metric C^h_{ij} serves as a priority score, where h indicates the recovery heuristic. Five graph-based heuristics and one random baseline were used.

Restoration strategies $h \in H$ are implemented as rank-based prioritizations of failed arcs using topological centrality measures. In addition, we incorporate static link travel times to identify shortest paths and use graph-based proxy variables to capture demand and the relative importance of junctions. For each disrupted arc (i, j), the centrality metric C^h_{ij} is computed and functions as a priority score. All disrupted arcs are then sorted according to this score and recovery actions are scheduled and implemented in the simulations in that order. In this study, we select six centrality measures that are commonly found in the literature in the context of transport infrastructure recovery decisions. In the remainder of this section, we define these centrality metrics.

Betweenness centrality

Betweenness centrality measures the degree to which a vertex or arc falls on the shortest path between other graph components. Using the previously introduced graph notation, betweenness centrality C^{B}_{ij} of an arc $(i, j) \in A$ can be defined as:

$$C^{\mathrm{B}}_{ij} = \sum_{r,s \in V, r \neq s, \sigma_{rs} > 0} \frac{\theta_{rs}(i,j)}{\theta_{rs}} \tag{1}$$

where θ_{rs} is the number of shortest paths from vertex r to vertex s, computed using link travel times σ_{ij} (minimizing total path travel times), and $\theta_{rs}(i, j)$ is the number of those shortest paths that pass through arc (i, j).

Degree centrality

Degree centrality measures the centrality of a vertex or arc based on the number of adjacent components. For a directed graph, the centrality of a vertex is split into incoming and outgoing arcs. Consider a vertex i, then the incoming degree centrality $C^{\mathrm{D,\ in}}_{v}(i)$ of this vertex is the number of incoming arcs and the outgoing degree centrality $C^{\mathrm{D,\ out}}_{v}(i)$ of this vertex is the number of outgoing arcs.

$$C^{\mathrm{D}}_{v_i} = \sum_{k \in V, (i,k) \in A} \alpha_{ik} + \sum_{k \in V} \alpha_{ki} \tag{2}$$

where α_{ik} and α_{ki} are the indicator variables corresponding to the presence of arcs (i, k) and (k, i), respectively. For arc (i, j), the degree centrality is then computed as the product of the outgoing centrality of the origin node and the incoming centrality of the destination node:

$$C^{\mathrm{D}}_{ij} = C^{\mathrm{D}}_{v_i} \cdot C^{\mathrm{D}}_{v_j} \tag{3}$$

Nearest-neighbour edge centrality

Near-neighbourhood centrality is a more recently developed edge centrality metric. Nearest-neighbour edge centrality could be an effective recovery strategy indicator because it can identify critical edges using local degree information, making it more efficient than global metrics under time-sensitive situations. For an arc (i, j), the near-neighbour edge centrality is defined as:

$$C^{\mathrm{NN}}_{ij} = \left(\frac{\rho_i + \rho_j - 2\sigma_{ij}}{|\rho_i - \rho_j| + 1} \right) \cdot \sigma_{ij}, \tag{4}$$

where ρ_i is the strength of vertex i expanded as $\rho_i = \sum_{j \in V} \sigma_{ij} + \sum_{j \in V} \sigma_{ji}$, and σ_{ij} is the travel time of arc (i, j).

The limitations of the nearest-neighbour edge centrality metric are that it does not account for the directionality of arcs, and it does not capture long-range structural dependencies in the network.

Closeness centrality

For closeness centrality, an edge is more central the shorter the paths are that connect it to all other edges. Let (i, j) be an arc between vertices i and j, then using the shortest path defined as σ_{ij} before, closeness centrality can be computed as:

$$C^{\mathrm{C}}_{ij} = (|A| - 1) \frac{C^{\mathrm{C}}_{v_i} C^{\mathrm{C}}_{v_j}}{C^{\mathrm{C}}_{v_i} + C^{\mathrm{C}}_{v_j}}; \quad C^{\mathrm{C}}_{v_i} = \frac{|V| - 1}{\sum_{k \in V} \sigma_{ki}} \tag{5}$$

where σ_{ki} is the shortest path distance from vertex k to vertex i, computed using travel time as edge weights (minimizing total weight), and $|V|$ is the number of vertices. If vertex s cannot reach some i, $\sigma_{ki} = \infty$, and if the case is true $\forall k \in V$, then $C^{\mathrm{C}}_{v_i} = 0$. $|A|$ is the number of arcs (edges), $C^{\mathrm{C}}_{v_i}$ and $C^{\mathrm{C}}_{v_j}$ are the closeness centality values of vertices i and j. If $C^{\mathrm{C}}_{v_i} + C^{\mathrm{C}}_{v_j} = 0$, set $C^{\mathrm{C}}_{ij} = 0$.

Eigenvector centrality

For eigenvector centrality, an edge is central if its adjacent edges are also central.

Let $\lambda_{\max}$ be the dominant eigenvalue of a matrix $\mathbf{M}$ where $\mathbf{M}$ is the edge adjacency matrix. $M_{(i,j)(k,\ell)} = 1$ if arcs (i, j) and (k, ℓ) are connected to the same node. Arc centrality is then computed as follows, through the power iteration method as a system of linear equations:

$$C^{\mathrm{EV}}_{ij} = \frac{1}{\lambda_{\max}} \sum_{(k,\ell)\in A} M_{(i,j)(k,\ell)} \sigma_{k\ell} C^{\mathrm{EV}}_{k\ell} \tag{6}$$

where A is the set of arcs, $M_{(i,j)(k,\ell)} = 1$ if arcs (i, j) and (k, ℓ) share a common vertex (considering directionality), else 0, $\sigma_{k\ell}$ is the travel time of arc (k, ℓ), and $\lambda_{\max}$ is the largest eigenvalue of the weighted edge adjacency matrix $\mathbf{M}$.

Supplementary Note 3: Resilience metrics and performance quantification

The multidimensional nature of road network resilience is captured through four aggregate performance indicators: operational efficiency, accessibility, spatial equity, and connectivity. All indicators are normalized to the undisrupted state ($P_m(t) \in [0, 1]$). The cumulative performance loss over recovery is:

$$\mathbf{P}_m = \frac{\int_{t_0}^{t_r} (1 - P_m(t))\, dt}{(t_r - t_0)} = \frac{\sum_{t=t_0}^{t_r-\Delta t} \left(1 - \frac{[P_m(t_i)+P_m(t_{i+1})]}{2}\right) \Delta t}{(t_r - t_0)}.$$

Smaller $\mathbf{P}_m$ values indicate higher resilience.

Operational efficiency

Operational efficiency refers to how efficiently people (and goods) can move between their origins and destinations. We define it as the average shortest-path travel time T_{total}, computed as:

$$P_{\text{Efficiency}}(t) = \frac{\text{Efficiency}(G(t))}{\text{Efficiency}(G)} \tag{7}$$

$$\text{Efficiency}(G(t)) = \frac{1}{n} \sum_{r=1}^{n} \sum_{\substack{s=1 \\ s\neq r}}^{n} \sigma_{rs}(t); \quad \sigma_{rs}(t) < \infty \tag{8}$$

where σ_{rs} is the shortest path travel time between nodes v_r and v_s, n is the number of vertices in G, and $\text{Efficiency}(G) = 1$. In this paper, we solely use topological data and do not weight the operational efficiency by commuter flows between OD pairs, though such weighting can be readily incorporated. A lower $P_{\text{Efficiency}}$ indicates lower resilience in terms of operational efficiency.

Accessibility

Accessibility refers to how easy it is to reach a node in the network. Here, we use closeness centrality as a proxy for accessibility. We define accessibility of a vertex v_i as A_i and compute it as the average closeness centrality-weighted reciprocals of travel times for each node, as follows:

$$P_{\text{Accessibility}}(t) = \frac{\text{Accessibility}(G(t))}{\text{Accessibility}(G} \tag{9}$$

$$\text{Accessibility}(G(t)) = \frac{1}{n} \sum_{i=1}^{n} \sum_{\substack{r=1 \\ r \neq i}}^{n} \frac{C_j}{\sigma_{ir}(t)}; \quad \sigma_{ir}(t) < \infty \tag{10}$$

where C_j is the closeness centrality of node r in the baseline network G. The pre-disruption closeness centrality of destination C_j is used as a surrogate measure for the economic and social opportunities at r accessible from i as those nodes which are central to the network may also have more opportunities (for example, central business district of a city). More severe disruptions result in lower values of $P_{Accessibility}$ values.

Spatial equity

Spatial equity refers to how fairly resources, services, or opportunities (in this case, transport accessibility) are distributed across a geographic area. In this study, we quantify spatial equity as the change in Moran's I corresponding to accessibility to opportunities at other nodes in the network. For every disrupted state of the network, we compute the node-level accessibility values and then compute their spatial autocorrelation using a weight matrix. The w spatial weights are computed as a function of Euclidean distances between nodes in the undisrupted state (Equation 12).

$$P_{\text{Spatial}}(t) = \max\left(0, \min\left(\frac{\text{Moran's } I(G(t))}{\text{Moran's } I(G)}, 1\right)\right) \tag{11}$$

$$\text{Moran's } I(G(t)) = \frac{n}{S_0} \cdot \frac{\sum_{r=1}^{n} \sum_{s=1}^{n} w_{rs}(t) \left(\mathcal{A}_r(t) - \overline{\mathcal{A}}(t)\right) \left(\mathcal{A}_s(t) - \overline{\mathcal{A}}(t)\right)}{\sum_{r=1}^{n} \left(\mathcal{A}_r(t) - \overline{\mathcal{A}}(t)\right)^2} \tag{12}$$

where

$$w_{rs}(t) = \begin{cases} \dfrac{1}{(d_{rs} + \epsilon)^b}, & r \neq s \\ 0, & r = s \end{cases}; \quad \mathcal{A}_r(t) = \sum_{\substack{i=1 \\ i \neq r}}^{n} \frac{C_i^{\text{C}}(0)}{\sigma_{ri}(t)}. \tag{13}$$

Here $\mathcal{A}_r(t)$ (for $\sigma_{ri}(t) < \infty$) is defined as accessibility of node r at time t, $\overline{\mathcal{A}}(t)$ is the mean accessibility across all nodes, $S_0 = \sum_{r=1}^{n} \sum_{s=1}^{n} w_{rs}$ is the sum of the spatial weights, d_{rs} is the Euclidean distance between v_r and v_s, ϵ is a small positive threshold added to avoid division by zero, and b is the distance decay parameter (a value of 1.0 km is used in this study). P_{Spatial} values are clipped between 0 and 1 as we are more interested in segregation of accessibility impacts which may result in disproportionate impact on certain affected regions during recovery.

Connectivity

Connectivity refers to how well different parts of the network are linked. We use the directed clustering coefficient following. For a node v_i, let k_i denote its total degree (the number of in- and out-neighbours) and $T_i(t)$ the number of directed triangles involving v_i at time t. The local directed clustering coefficient is defined as

$$C_i(G(t)) = \frac{T_i(t)}{k_i\,(k_i - 1)}.$$

We compute network-level connectivity as the average of these local coefficients across all n nodes:

$$\text{Connectivity}(G(t)) = \frac{1}{n} \sum_{i=1}^{n} C_i(G(t)).$$

Finally, we define the connectivity performance metric as

$$P_{\text{Connectivity}}(t) = \frac{\text{Connectivity}(G(t))}{\text{Connectivity}(G)}.$$

Supplementary Note 4: Simulation framework and pseudocode

The stress-testing experiments for data generation are implemented through a structured simulation framework that applies disruptions to each network, restores links step by step according to different strategies, and evaluates multidimensional resilience outcomes. The main steps of this procedure are outlined in Algorithm 1.

First, for each of the selected road networks, we generate disruptions by randomly removing a specified percentage of total edges in the network and evaluate the resulting performance relative to the undisrupted network. We adopt random incremental failures ((10%, 20%,. . . ,100% road links are failed) as a hazard-agnostic stress-testing approach. This provides a neutral baseline that enables comparison of resilience outcomes across a large and diverse set of cities without bias toward any specific hazard. To address the stochasticity of disruptions and subsequent recovery, we repeat each simulation 25 times for each combination of initial disruption percentage and recovery strategy within a network. Random failures are widely used in network science as a benchmark for robustness analysis, and they capture a broad spectrum of possible disruption patterns, including unforeseen or compound failures. Second, for each disruption scenario and recovery strategy, edges are restored step by step according to strategy rankings, with performance metrics (efficiency, accessibility, spatial equity, and connectedness) recorded after each step and recovery curves are derived. We use a predefined percentage of total links in the network to define the amount of repair actions in each time-step. In this way, we ensure that the amount of recovery resource available in a network is proportional to its size and thus does not influence the recovery outcomes. These curves are conceptually aligned with the resilience triangle framework, representing system performance loss and recovery over time. Third, the resulting recovery curves are aggregated across all networks, disruption levels, and strategies, and resilience metrics are derived by applying trapezoidal integration to measure the area above the curves. Finally, these resilience outcomes are combined with network-level topological variables to create a comprehensive dataset, which is then used to estimate beta regression models and quantify the influence of topology, disruption intensity, and recovery strategy on multidimensional resilience.

Algorithm 1 Pseudocode for simulation workflow implementation and final dataset generation

```
Input: Set of road networks G ∈ 𝐆, recovery strategies h ∈ H, damage percentages pct ∈ 𝒫, iterations N, incremental restoration percentage τ
Output: Simulation dataset for regression analysis
for network G ∈ 𝐆 do
    for damage percentage pct ∈ 𝒫 do
        for disrupted edge combination i = 1 to N do
            Apply disruption: remove pct% of edges to create disrupted network G(0)
            Evaluate baseline performance on G
            Evaluate performance on G(0)
            for strategy h ∈ H do
                Initialize restoration counter t ← 1
                while edges remain to restore do
                    Select edges to restore in this iteration based on ranking from h
                    Restore edges and update disrupted edge list
                    Re-evaluate network performance of (partially) restored network G(t) across resilience dimensions m ∈ M and compute corresponding normalized metrics P_m(t)
                    t ← t + 1
                end while
            end for
        end for
    end for
    Aggregate results across strategies, iterations, and damage levels for G
end for
Combine results across all networks into unified dataset
Compute topology-related variables for all networks
Derive resilience metrics using trapezoidal integration of recovery curves
Merge variables corresponding to topological, disruption, strategy, and resilience metrics into final dataset
```

Supplementary Note 5: Statistical modeling of recovery outcomes

To assess the influence of topological features of networks on the performance of recovery strategies, a statistical analysis is conducted using beta-regression. This model is selected due to the bounded [0,1] scale of the performance metrics $\mathbf{P}_m$, which quantify the effectiveness of recovery strategies as the area above the recovery curves. The dependent variable in the analysis is one of the performance metrics defined in Supplementary Note 3.

The general specification of the regression model is given in Equation 14. A logit transformation is applied, where $\eta_m = \ln\left(\mu_m/(1-\mu_m)\right)$ and μ_m denotes the expected value of the standardized performance metric $\mathbf{P}_m$. Coefficients $(\beta_{0,m}, \beta_{1,m}, \delta_{r,m}, \gamma_{h,m}, \phi_{h,m})$ quantify the influence of the predictors: *pct* represents the proportion of disrupted links (damage level), x_r denotes the topological properties, $\mathbb{I}(h)$ indicates recovery strategies relative to the baseline (random recovery), and $\phi_{h,m}$ captures the interaction between recovery strategy and damage level.

$$\eta_m = \beta_{0,m} + \beta_{1,m} pct + \sum_{r \in R} \delta_{r,m} x_r + \sum_{h \in S} \gamma_{h,m} \cdot \mathbb{I}(h) + \sum_{h \in S} \phi_{h,m}(\mathbb{I}(h) \cdot pct), \tag{14}$$

The predictors included in this model fall into four categories: (i) the baseline damage variable *pct*, (ii) structural features of the undisturbed network, (iii) categorical indicators for the recovery strategy applied, and (iv) interaction terms between damage and recovery strategy.

The topological features were selected from an initial set of network characteristics that capture various structural properties, such as size, connectivity, redundancy, and spatial embedding (number of edges, density, average and variability of node degrees, global and average clustering coefficients, assortativity by degree, average link length (metres), and average link travel time (minutes).

Model fitting is performed via maximum likelihood estimation using the `betareg` package in R. Continuous variables are standardized before estimation. Model diagnostics (residual plots, likelihood ratio tests) are used to test model robustness and goodness of fit.

Data and Code Availability

All simulation scripts, processed datasets, and analysis code are available on the GitHub repository https://github.com/srijithbalakrishnan/transport_stresstesting. The road network data are publicly accessible through the `OpenStreetMap` platform under the Open Database License (ODbL).