Data in Brief 16 (2018) 1069–1077

Contents lists available at ScienceDirect

Data in Brief

journal homepage: www.elsevier.com/locate/dib

Data Article

Multilayered complex network datasets for three supply chain network archetypes on an urban road grid

Nadia M. Viljoen [*], Johan W. Joubert

Centre for Transport Development, Department of Industrial and Systems Engineering, University of Pretoria, 0002 South Africa

ARTICLE INFO

Article history:
Received 2 October 2017
Accepted 15 December 2017
Available online 21 December 2017

Keywords:
Multilayered complex networks
Supply chain vulnerability
Urban road networks

ABSTRACT

This article presents the multilayered complex network formulation for three different supply chain network archetypes on an urban road grid and describes how 500 instances were randomly generated for each archetype. Both the supply chain network layer and the urban road network layer are directed unweighted networks. The shortest path set is calculated for each of the 1 500 experimental instances. The datasets are used to empirically explore the impact that the supply chain's dependence on the transport network has on its vulnerability in Viljoen and Joubert (2017) [1]. The datasets are publicly available on Mendeley (Joubert and Viljoen, 2017) [2].

Specifications Table

Subject area	*Complex Network Theory, Logistics and Supply Chain Management, Transport Planning*
More specific subject area	*Multilayered Complex Networks, Supply Chain Vulnerability, Urban Road Networks*
Type of data	*Extensible Markup Language (XML) files, compressed.*

[*] Corresponding author.
E-mail addresses: nadia.viljoen@up.ac.za (N.M. Viljoen), johan.joubert@up.ac.za (J.W. Joubert).

http://dx.doi.org/10.1016/j.dib.2017.12.041

How data was acquired	*The 500 random instances for each of the three supply chain network archetypes were generated using the JUNG library in Java. The shortest path sets were calculated using Dijkstra's algorithm and the JUNG library in Java.*
Data format	*Raw*
Experimental factors	*N/A*
Experimental features	*The datasets were used as input to an iterative disruption simulation that explores the impact that a supply chain's dependence on the urban road network has on its vulnerability.*
Data source location	*Not applicable. Data are randomly generated.*
Data accessibility	*The data are available in this article and publicly on Mendeley* [2]

Value of the data

- The dataset can be used to empirically test a number of complex network phenomena such as targeted attack & random error, spreading and synchronisation on three supply chain network archetypes layered on the urban road grid.
- The shortest path sets open up opportunities for research pertaining to supply chain optimisation and city planning.
- The dataset is a first example of a multilayered complex network that combines a supply chain (logical layer) with an urban road network (physical layer) and can be used as a prototype for modelling other parts of the supply chain.

1. Data

The data accompanying this article include the Extensible Markup Language (XML) files required to construct the 1500 unique multilayered complex network instances (500 instances for each of three supply chain archetypes). Each XML file contains all the information pertaining to one specific multilayered instance. It also contains the shortest path sets calculated for each instance. All datasets are publicly available from Mendeley [2]. The detailed Document Type Definition (DTD), which contains the declarations that describes the formal acceptable structure of the XML file, is available on http://www.matsim.org/files/dtd/multilayerNetwork_v1.dtd.

2. Experimental design, materials and methods

2.1. Data definitions and mathematical formulation

The generic multilayered network formulation is based on the notation presented in [3]. This formulation is then adapted for this specific scenario where one layer represents the supply chain network and the other the urban road grid.

2.2. Generic multilayered formulation

The *multilayer network* is a pair $\mathcal{M} = (\mathcal{G}, \mathcal{C})$ where $\mathcal{G} = \{G_m; m \in \{1, \ldots, M\}\}$ is a family of M individual graphs $G_m = (X_m; E_m)$ which each represent a layer of $\mathcal{M}$. In the generic formulation, as presented in [1], α and β refer to layers of $\mathcal{G}$ such that $\alpha, \beta \in 1, 2, \ldots, M$ and $\alpha \neq \beta$. The set of nodes in layer G_α are denoted by $X_\alpha = x_1^\alpha, \ldots, x_{N_\alpha}^\alpha$ where N_α is the number of nodes in G_α. The edges are denoted by $E_\alpha \subseteq X_\alpha ñ X_\alpha$. The set of interconnections between nodes in G_α and G_β with $\alpha \neq \beta$ are defined by

$$\mathcal{C} = E_{\alpha,\beta} \subseteq X_\alpha ñ \quad X_\beta; \alpha, \beta \in \{1, 2, \ldots, M\}, \alpha \neq \beta$$

Therefore the elements of $E_{\alpha,\beta}, \alpha\neq\beta$ are *interlayer* connections while elements of E_α are the *intralayer* connections.

2.3. Scenario-specific formulation

As presented in [1], we adapt the generic formulation to improve readability. Indices referring to layers are superscripts instead of subscripts so as not to cause confusion with node indices. We let $\mathcal{M}=(\mathcal{G},\mathcal{C})$ be the multilayered network where $\mathcal{G}=\left(G^{1K},G^2\right)$. The first layer $\left(G^{1K}\right)$ represents the supply chain network where the nodes represent logistics facilities and the links represent the relationships between these facilities based on the movement of freight. This is also referred to as the *logical* layer. The second layer (G^2) represents the urban road network and is thus a *physical* layer.

In G^{1K}, K denotes the supply chain network archetype with $K\in\{F,S,D\}$ where F is the Fully Connected (FC) network, S is the Single Hub (SH) network and D is the Double Hub (DH) network. The node set of G^{1K} is defined as:

$$N^{1K}=12$$

$$X^{1K}=\left\{x_1^{1K},\ldots,\quad x_{N^{1K}}^{1K}\right\}\forall K\in F,S,D$$

and the edges by:

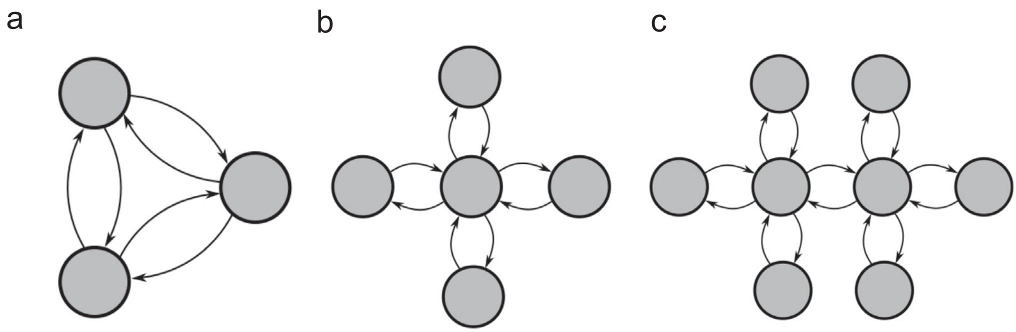
$$E^{1K}=\left\{e_{ij}^{1K}\right\}\forall i\in\left\{1,\ldots,N^{1K}\right\},j\in\left\{1,\ldots,N^{1K}\right\}\text{ and } i\neq j$$

where

$$e_{ij}^{1K}=\begin{cases}1, & \text{if } x_i^{1K}\text{is connected to } x_j^{1K}\\ 0, & \text{otherwise}\end{cases}\quad\forall K\in F,S,D$$

An FC network, $G^{1F}=\left(X^{1F},E^{1F}\right)$, assumes that all nodes in the network are directly connected to all other nodes so that each node x_i^{1F} is directly connected to every other node x_j^{1F} by e_{ij}^{1F} where $i\neq j$. Fig. 1a) shows an example of G^{1F} with three nodes. An SH network, $G^{1S}=\left(X^{1S},E^{1S}\right)$, assumes that there is one hub node with all other nodes connecting directly to the hub but not to one another as illustrated in Fig. 1b). A DH network, $G^{1D}=\left(X^{1D},E^{1D}\right)$, assumes that there are two hub nodes with half of the remaining nodes connected directly to the first hub and the other half connected directly to the second as illustrated in Fig. 1c). G^{1K} is a directed, unweighted network for all $K\in\{F,S,D\}$.

In [1] each of the supply chain network archetypes have 12 nodes (i.e. $N^{1F}=N^{1S}=N^{1D}=12$). Therefore the Single and Double Hub archetypes both have 22 edges in $\left|E^{1S}\right|$ and $\left|E^{1D}\right|$ while the Fully Connected archetype has $\left|E^{1F}\right|=N^{1F}\left(N^{1F}-1\right)=132$.



Fully Connected (FC) archetype G^{1F} with N^{1F}=3.

Single Hub (SH) archetype G^{1S} with N^{1S}=5.

Double Hub (DH) archetype G^{1D} with N^{1D}=8.

Fig. 1. Examples of the three supply chain network archetypes (reproduced from [1]).

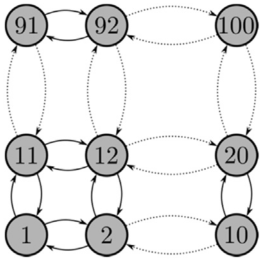


Fig. 2. Graphic representation of the urban road grid $\boldsymbol{G^2}$ (adapted from [1]).

G^2 represents the urban road network and is a simplified grid network with 100 intersections organised in a 10 × 10 configuration as shown in Fig. 2. The node set is defined by:

$$N^2 = 100$$

$$X^2 = \left\{x_1^2, \ldots, \quad x_{N^2}^2\right\} \forall K \in F, S, D$$

and the edges by:

$$E^2 = \{e_{st}^2\} \forall s, t \in \left\{1, 2, \ldots, N^2\right\} \text{and } s \neq t$$

where

$$e_{st}^2 = \begin{cases} 1, & \text{if } x_s^2 \text{ is connected to } x_t^2 \\ 0, & \text{otherwise} \end{cases}$$

This is also a directed, unweighted network and thus nodes are connected with two directed, opposite arcs, ⇆, instead of one undirected edge, ⟷. The assumption is that when a road segment in one direction fails, the associated lane in the opposite direction is not necessarily affected.

In $G^2 = (X^2, E^2)$ each node x_s^2 is connected to four neighbours unless it is on the boundaries of the grid, in which case it is only connected to three neighbours or on the corners of the grid in which case it has only two neighbours.

2.3.1. *Associating the network layers*

To create an instance of the multilayered network $\mathcal{M}$ the interlayer connections, $E^{1K,2}$, need to be defined. This is done by associating each node $x_i^{1K} \in X^{1K}$ with a node $x_s^2 \in X^2$. The simplifying assumption is made that every logistics facility corresponds to the intersection closest to it. Any grid node x_s^2 may be associated with at most one node in X^{1K}. The *interlayer* adjacency matrix is denoted by $A^{[1K,2]} = (a_{is}^{1K,2})$, where.

$$a_{is}^{1K,2} = \begin{cases} 1, & \text{if} (x_i^{1K}, x_s^2) \in E^{1K,2} \forall i \in 1, 2, \ldots, N^{1K}, s \in 1, 2, \ldots, N^2 \\ 0, & \text{otherwise} \end{cases}$$

The pseudocode below (Algorithm 1) shows how the associations are randomly generated to produce $A^{[1F,2]}$.

Algorithm 1:. Random generation of $A^{[1F,2]}$

```
Input:    $G^{1F}$, $G^{2}$
Output:   $A^{[1F,2]}$
$used \leftarrow NULL$ // vector that stores $x_i^2$already assigned;
$A^{[1F,2]} \leftarrow NULL$ // set all elements in matrix as unassigned;
for $x_i^{1F} \in X^{1F}$do
    $continue$ =TRUE;
    while $continue$ do
        $randVertex$ ←randomly selected $x_j^2 \in X^2$;
        if $randVertex \notin used$ then
            $a^{[1F,2]} \leftarrow 1$;
            Append $used$ with $randVertex$;
            $continue$ = FALSE;


return $A^{[1F,2]}$
```

The algorithm used to generate $A^{[1S,2]}$ is similar and shown in the pseudocode below (Algorithm 2).

Algorithm 2:. Random generation of $A^{[1S,2]}$

```
Input:    $G^{1S}$, $G^{2}$
Output:   $A^{[1S,2]}$
$used \leftarrow NULL$ // vector that stores $x_i^2$already assigned;
$A^{[1S,2]} \leftarrow NULL$ // set all elements in matrix as unassigned;
for $x_i^{1S} \in X^{1S}$do
    $continue$ =TRUE;
    while $continue$ do
        $randVertex$ ←randomly selected $x_j^2 \in X^2$;
        if $randVertex \notin used$ then
            $a^{[1S,2]} \leftarrow 1$;
            Append $used$ with $randVertex$;
            $continue$ = FALSE;


return $A^{[1S,2]}$
```

Finally the algorithm used to generate $A^{[1D,2]}$ is slightly different as it has to constrain the assignment of nodes to hubs. The two hubs are first associated with grid nodes. Thereafter, a node is assigned to a hub if and only if the shortest path distance along G^2 from that node to the hub is less than or equal to the distance from that node to the other hub. The pseudocode below (Algorithm 3) presents the algorithm.

Algorithm 3:. Random generation of $A^{[1D,2]}$

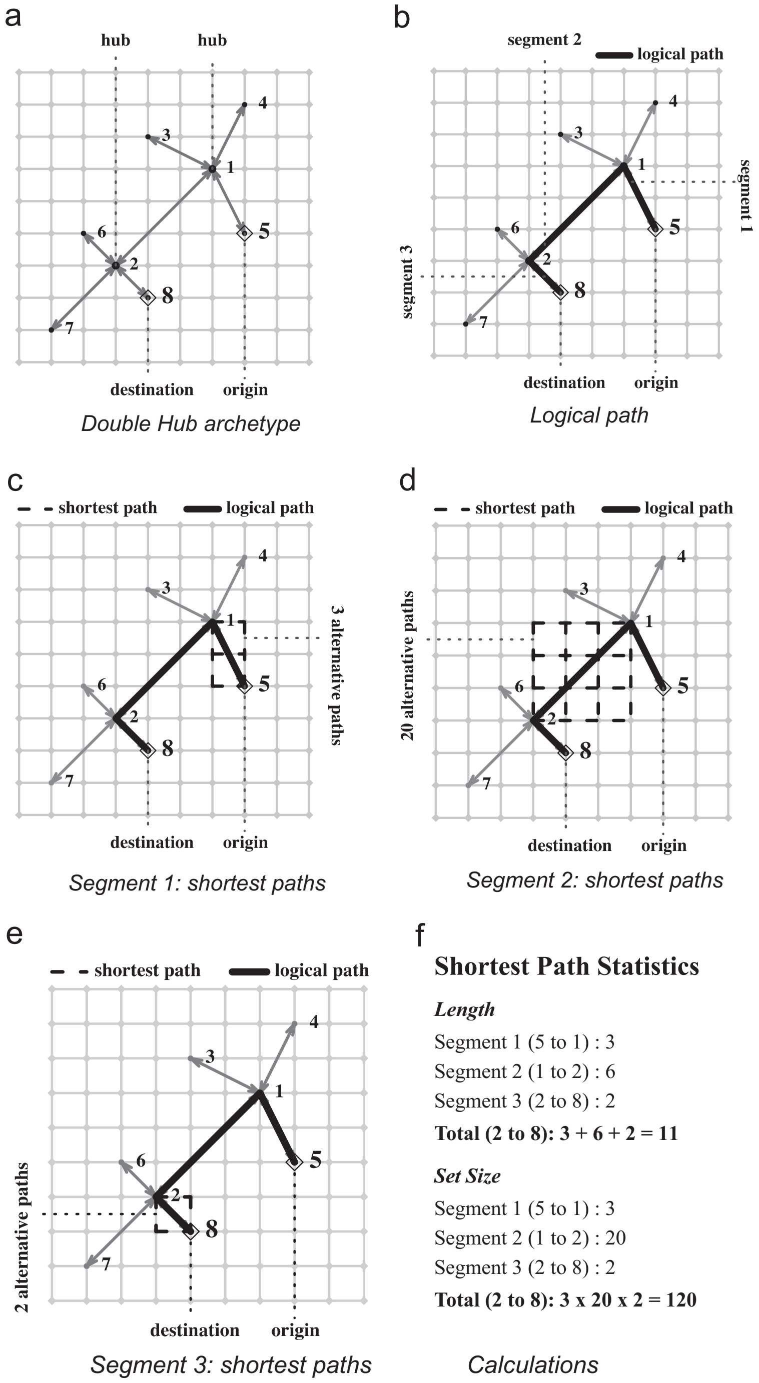
```
Input:    $G^{1D}, G^2$
Output:   $A^{[1D,2]}$
$used \leftarrow NULL$ // vector that stores $x_i^2$already assigned;
$A^{[1D,2]} \leftarrow NULL$ // set all elements in matrix as unassigned;
// Assign hubs first
for $i \leftarrow 1$ to 2 do
    $continue$ =TRUE;
    while $continue$ do
        $randVertex$ ←randomly selected $x_j^2 \in X^2$;
        if $randVertex \notin used$ then
            $a^{[1D,2]} \leftarrow 1$;
            Append $used$ with $randVertex$;
            $continue$ = FALSE;

// Assign remaining nodes
for $i \leftarrow 3$ to $N^{1D}$ do
    $continue$ =TRUE;
    while $continue$ do
        $randVertex$ ←randomly selected $x_j^2 \in X^2$;
        if $randVertex \notin used$ then
            $Dist1$ ← Dijkstra's shortest path on $G^2$ between $randVertex$ and $x_1^{1D}$;
            $Dist2$ ← Dijkstra's shortest path on $G^2$ between $randVertex$ and $x_2^{1D}$;
            if $i \leq 2 + (N^{1D} - 2)/2$ then // vertices around hub 1//
                if $Dist1 \leq Dist2$ then //distance to hub 1 must be smaller than to hub 2
                    $a^{[1D,2]} \leftarrow 1$;
                    Append $used$ with $randVertex$;
                    $continue$ = FALSE;
            else //vertices around hub 2//
                if $Dist2 \leq Dist1$ then //distance to hub 2 must be smaller than to hub 1
                    $a^{[1D,2]} \leftarrow 1$;
                    Append $used$ with $randVertex$;
                    $continue$ = FALSE;

return $A^{[1D,2]}$
```

The experiments conducted in [1] required large samples of $\mathcal{M}$ for each of the three archetypes. We therefore generated 500 instances for each archetype with $N^{1K} = 12 \forall K \in F, S, D$. We did not explicitly prevent the generation of identical $A^{[1K,2]}$ as the likelihood was negligible.

The data required to reconstruct each instance is contained in an XML file. The datafile includes the following:

1. Node set and edgelist of G^2;
2. Node set and edgelist of G^{1K};
3. The association between nodes in X^{1K} and X^2; and
4. The shortest path sets unique to each instance (discussed in the next section).

A template of the .xml file structure is shown in Fig. 4.



Double Hub archetype

Logical path

Segment 1: shortest paths

Segment 2: shortest paths

Segment 3: shortest paths

Shortest Path Statistics

Length

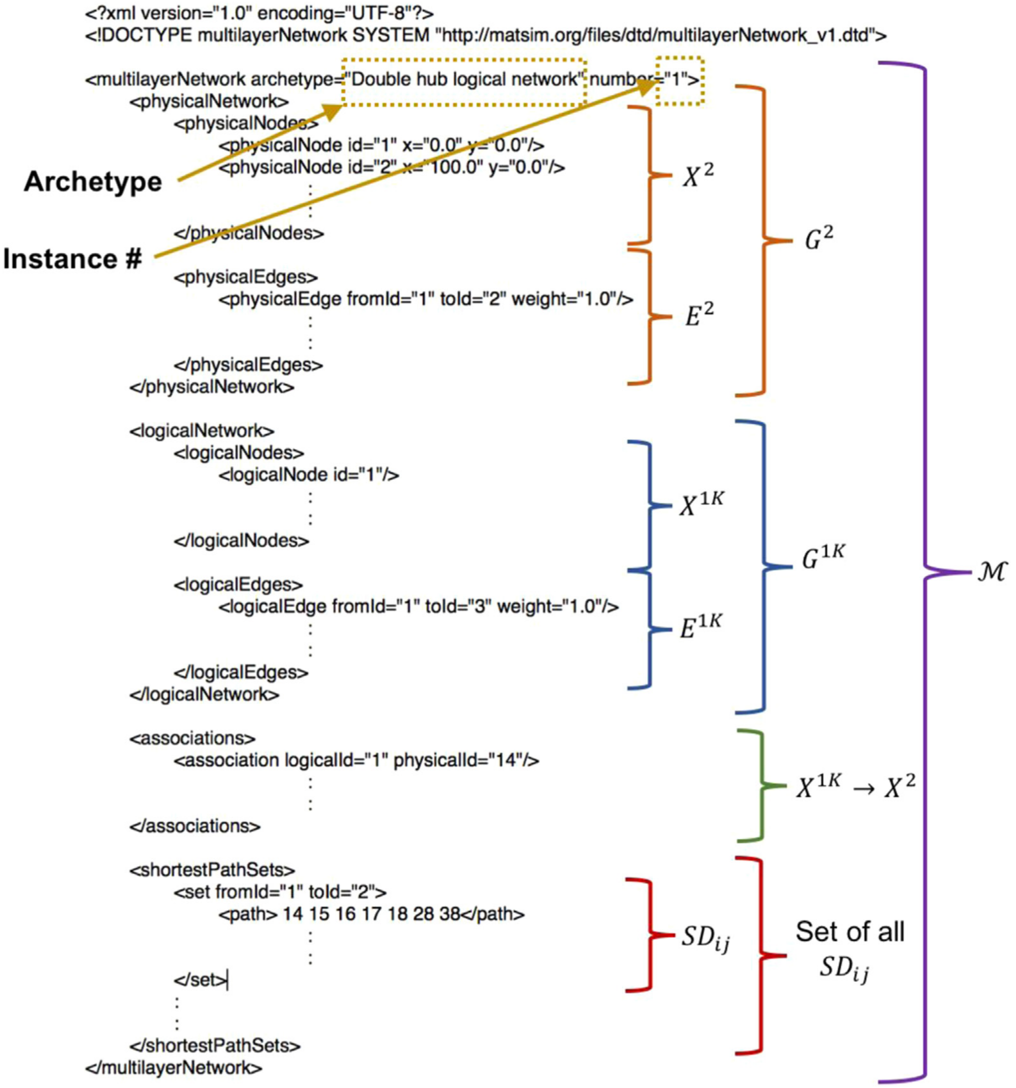
Segment 1 (5 to 1) : 3
Segment 2 (1 to 2) : 6
Segment 3 (2 to 8) : 2
Total (2 to 8): 3 + 6 + 2 = 11

Set Size

Segment 1 (5 to 1) : 3
Segment 2 (1 to 2) : 20
Segment 3 (2 to 8) : 2
Total (2 to 8): 3 x 20 x 2 = 120

Calculations

Fig. 3. Example of calculating a shortest path set on $\mathcal{M}$ (reproduced from [1]).



```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE multilayerNetwork SYSTEM "http://matsim.org/files/dtd/multilayerNetwork_v1.dtd">

<multilayerNetwork archetype="Double hub logical network" number="1">
    <physicalNetwork>
        <physicalNodes>
            <physicalNode id="1" x="0.0" y="0.0"/>
            <physicalNode id="2" x="100.0" y="0.0"/>
                :
                :
        </physicalNodes>

        <physicalEdges>
            <physicalEdge fromId="1" toId="2" weight="1.0"/>
                :
                :
        </physicalEdges>
    </physicalNetwork>

    <logicalNetwork>
        <logicalNodes>
            <logicalNode id="1"/>
                :
                :
        </logicalNodes>

        <logicalEdges>
            <logicalEdge fromId="1" toId="3" weight="1.0"/>
                :
                :
        </logicalEdges>
    </logicalNetwork>

    <associations>
        <association logicalId="1" physicalId="14"/>
                :
                :
    </associations>

    <shortestPathSets>
        <set fromId="1" toId="2">
            <path> 14 15 16 17 18 28 38</path>
                :
                :
        </set>
        :
        :
    </shortestPathSets>
</multilayerNetwork>
```

Fig. 4. Template of the data file structure for an instance $\mathcal{M}$.

2.3.2. Calculating shortest path sets

The shortest path sets of an instance of $\mathcal{M}$is its unique fingerprint as it is a function of the random association of X^{1K} to X^2. The characteristics of these shortest path sets are examined in [1]. Apart from the fact that these sets characterise $\mathcal{M}$, they are computationally cumbersome to calculate. For these two reasons the sets are included in the input data.

To explain how the shortest path sets were calculated, we use the example of a Double Hub instance of $\mathcal{M}$ with $N^{1D} = 8$. Suppose X^{1D} was placed on X^2 as shown in Fig. 3a) and the shortest path had to be calculated between origin x_5^{1D} and destination x_8^{1D} as indicated. The logical shortest path on G^{1D} consists of three segments and is highlighted in Fig. 3b). However, freight travelling between x_5^{1D} and x_8^{1D} must make use of the road network and is therefore also constrained by G^2. Fig. 3c) shows that there are three alternative shortest paths, each of length three, corresponding to segment one of the logical path. Similarly, there are 20 alternative paths of length six for segment two (Fig. 3d) and two alternative paths of length two for segment three (Fig. 3e). The length of the shortest path on $\mathcal{M}$ is the sum of the lengths of the shortest paths for each segment ($3+6+2=11$) while the total number of shortest paths is the product of the number of paths for each segment ($3*20*2=120$) (Fig. 3f). The shortest path set between x_5^{1D} and x_8^{1D} in $\mathcal{M}$ is the collection of the 120 unique path sequences and is denoted by$\mathcal{S}_{58}$.

Metrics pertaining to shortest path sets refer to a specific instance of $\mathcal{M}$, therefore the subscripts relating to the layers and supply chain archetype are dropped for simplicity's sake. Generally we define:

$$\mathcal{S}_{\boldsymbol{ij}} = SD_{ij}, SI_{ij}$$

where SD_{ij} is the subset of all shortest path sets between node-pairs that are directly connected in G_1^K such that:

$$SD_{ij} = \{s^1, s^2, \ldots, s^{P_{ij}}\} \quad \forall \quad x_i^{1K}, x_j^{1K} \in E^{1K}$$

and $SI_{ij}$is the subset of all the shortest path sets between node-pairs that are indirectly connected in G^{1K} such that:

$$SI_{ij} = \{s^1, s^2, \ldots, s^{P_{ij}}\} \quad \forall \quad x_i^{1K}, x_j^{1K} \notin E^{1K}$$

and where P_{ij} is the number of alternative shortest paths between any node-pair.

Enumerating and storing $\mathcal{S}_{ij}$ for all $K \in (F, S, D); i, j \in \left(1, \ldots, N^{1K}\right); i \neq j$ is too cumbersome. Instead we only calculate and store SD_{ij}. In the example of Fig. 3 that would mean we only calculate and store SD_{51}, SD_{12} and SD_{28} but not SI_{58} as this set can always be constructed from the others when required. We calculate the shortest path sets using Dijkstra's algorithm in Java.

Acknowledgements

Our gratitude to the National Research Foundation of South Africa (Grant UID: 105519) for partially funding the doctoral research that led to this article.

Transparency document. Supporting information

Supplementary data associated with this article can be found in the online version at http://dx.doi.org/10.1007/s11067-017-9370-1.

References

[1] N.M. Viljoen and J.W. Joubert, Networks and Spatial Economics, 2017. http://dx.doi.org/10.1007/s11067-017-9370-1.

[2] J.W. Joubert and N.M. Viljoen, Multilayer complex networks, v3. Mendeley Data, 2017. Available online from http://dx.doi.org/10.17632/268byhmvv5.3.

[3] S. Boccaletti, G. Bianconi, R. Criado, C.I. Del Genio, J. Gómez-Gardeñes, M. Romance, I. Señdina-Nadal, Z. Wang, M. Zanin, The structure and dynamics of multilayer networks, Phys. Rep. 544 (1) (2014) 1–122.