Computational Social Networks
a SpringerOpen Journal

RESEARCH

CrossMark

Optimization problems in correlated networks

Song Yang, Stojan Trajanovski and Fernando A. Kuipers[*]

*Correspondence:
F.A.Kuipers@tudelft.nl
Faculty of Electrical
Engineering, Mathematics
and Computer Science, Delft
University of Technology,
P.O. Box 5031, 2600 GA Delft,
The Netherlands

Abstract

Background: Solving the shortest path and min-cut problems are key in achieving high-performance and robust communication networks. Those problems have often been studied in deterministic and uncorrelated networks both in their original formulations as well as in several constrained variants. However, in real-world networks, link weights (e.g., delay, bandwidth, failure probability) are often correlated due to spatial or temporal reasons, and these correlated link weights together behave in a different manner and are not always additive, as commonly assumed.

Methods: In this paper, we first propose two correlated link weight models, namely (1) the deterministic correlated model and (2) the (log-concave) stochastic correlated model. Subsequently, we study the shortest path problem and the min-cut problem under these two correlated models.

Results and Conclusions: We prove that these two problems are NP-hard under the deterministic correlated model, and even cannot be approximated to arbitrary degree in polynomial time. However, these two problems are solvable in polynomial time under the (constrained) nodal deterministic correlated model, and can be solved by convex optimization under the (log-concave) stochastic correlated model.

Keywords: Shortest path, Min-cut, Correlated networks, Stochastic link weights

Background

Both the shortest path problem and the min-cut problem are of great importance to various kinds of network routing applications (e.g., in transportation networks, optical networks, etc.). A traffic request can be routed in the most efficient way (e.g., with minimum delay) by computing a shortest path. On the other hand, the min-cut problem arises in the context of network reliability, network throughput, etc. Fortunately, both of these problems are solvable in polynomial time for networks with independent additive link weights.

However, often correlations or (inter-)dependencies exist among link weights. For example, in overlay [1] or multilayer networks [2], the abstract links in the logical layer are mapped to different links in the physical layer. In this context, two or more abstract links, which use the same physical links, may have correlated latencies [3], bandwidth usage [4], or geographical failures [5, 6]. Or if the path must pass through some specific nodes (e.g., regenerators to boost the signal quality [7]), such important nodes and their

Yang *et al. Compu Social Networks* (2016) 3:1

Page 2 of 20

links may also introduce dependencies. Correlations also appear in social networks. For instance, a message may be forwarded more rapidly if it came from a close friend rather than from a distant acquaintance. Another example relates to interdependent networks [8], where for instance the electricity network and Internet are coupled and inter-connected, and one node or link failure in one network may cause failures of nodes or links in the other network. Similarly, in Shared-Risk Link Group (SRLG) networks [9], links in, for example, the same duct will fail simultaneously, if their duct fails. The dependencies in interdependent and SRLG networks can also be seen as correlations, so we use the term correlation throughout this paper and study relevant problems in these so-called correlated networks. Our key contributions are as follows:

- We propose two correlated link weight models, namely a *deterministic correlated model* and a *stochastic correlated model*.
- We study the shortest path problem and the min-cut problem under the deterministic correlated model, and we prove that both of them are NP-hard and even cannot be approximated in polynomial time.
- On the other hand, we also show that both the shortest path problem and the min-cut problem are solvable in polynomial time under a (constrained) nodal deterministic correlated model.
- To solve both problems under the proposed correlated models, we propose exact algorithms under the deterministic correlated model, and develop convex optimization formulations for the stochastic correlated model.

The remainder of this paper is organized as follows. "Correlated link weight models" section introduces our two correlated link weight models. In "Shortest paths in correlated networks" and "Minimum cuts in correlated networks" sections, we study the shortest path problem and min-cut problem, respectively, for the proposed models and devise algorithms to solve them exactly. An overview of the related work is presented in "Related work" section and we conclude in "Conclusions" section.

Correlated link weight models

A network having node and link weights can be transformed into a directed network with only link weights, as done in [10]. Therefore, we assume that nodes are unweighted and only consider correlated link weights. Throughout this paper, we use the term "correlated model" to represent "correlated link weight model."

Deterministic correlated model

Without loss of generality, we use $w(l)$ to represent the weight of link l. For simplicity, in this paper we call $w(l)$ the cost of l, although it could also reflect other metrics such as delay and energy. In the deterministic correlated model, for any two links l_i and l_j, their joint total cost is represented by $w(l_i) \oplus w(l_j)$, where the operator $\oplus$ indicates the joint total cost of the links, which may differ from the $+$ operator when they are *correlated*. When correlated, the use of one link may influence the cost of another in this model. For example, in Fig. 1 where the cost is shown above each link, it is assumed that only links (s, a) and (b, t) are correlated with joint cost of 11, and all the other links

Yang *et al. Compu Social Networks* (2016) 3:1

Page 3 of 20

Fig. 1 An example of the deterministic correlated model
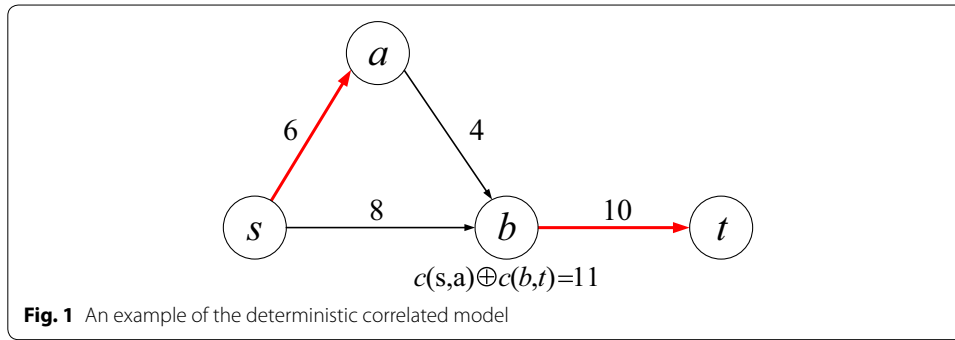
have uncorrelated costs. We can see that in path s–b–t, the cost of link (b, t) is 10, since another link (s, b) in this path is not correlated with it. Therefore, this path's cost is equal to 18. However, in path s–a–b–t, the cost of link (b, t) should be calculated together with link (s, a), leading to a joint cost 11, since they are correlated and both appear in this path. Therefore, this path's total cost is equal to $11 + 4 = 15$, which is smaller than the sum of the individual link costs ($6 + 4 + 10 = 20$).

Equivalently, we could formulate $w(l_i) \oplus w(l_j) = \rho_{i,j} \cdot (w(l_i) + w(l_j))$, where $\rho_{i,j}$ stands for the correlation coefficient between links l_i and l_j, and its value varies in the range of $(0, \infty)$, since we do not consider negative costs. When $\rho_{i,j}$ is equal to 1, l_i and l_j are uncorrelated, when $\rho_{i,j}$ is greater than 1, l_i and l_j have an increasing correlation, and otherwise we say that l_i and l_j have a decreasing correlation.

Analogously, for given $m > 1$ links $l_1, l_2, \ldots, l_m$ in the deterministic correlated model, their joint total cost can be expressed as follows:

$$w(l_1) \oplus w(l_2) \cdots \oplus w(l_m) = \rho_{1,2,\ldots,m} \cdot (w(l_1) + w(l_2) + \cdots + w(l_m)). \tag{1}$$

Similarly, if the link l's weight is multiplicative (e.g., failure probability), then by using $-log(w(l))$ to represent its weight value, Eq. (1) also applies. The decreasing correlation case can also reflect SRLG networks. For instance, in SRLG networks, each link is associated with several SRLG events with their respective failure probabilities. Hence, the total failure probability (represented by P_{SRLG}) of two correlated links that have at least one SRLG in common will be equal to the product of the failure probabilities of all the distinct SRLG events that belong to these two links. Let us denote by $P_{l_1} = P_{s_1} \cdot P_s$ and $P_{l_2} = P_{s_2} \cdot P_s$ the failure probabilities of these two links, respectively, where P_s denotes the common SRLGs' failure probability between l_1 and l_2, and P_{s_1} (P_{s_2}) is the non-overlapping SRLGs' failure probability of l_1 (l_2). Then $P_{l_1} \cdot P_{l_2} < P_{SRLG} = P_{s_1} \cdot P_{s_2} \cdot P_s < \min(P_{l_1}, P_{l_2})$. By taking the $-log$, we have

$$\max\left(-\log(P_1), -\log(P_2)\right) < -\log(P_{SRLG}) < \left(-\log(P_{l_1}) + (-\log(P_{l_2}))\right).$$

Or equivalently,

$$-\log(P_{SRLG}) = \rho \cdot \left(-\log(P_{l_1}) + (-\log(P_{l_2}))\right), \tag{2}$$

where $\rho < 1$ denotes their correlation coefficient.

In probability theory, given two random variables X and Y with expected values μ_X and μ_Y, and standard deviations σ_X and σ_Y, their *linear* correlation coefficient $\rho(X, Y)$ is defined as

Yang *et al. Compu Social Networks* (2016) 3:1

Page 4 of 20

$$\rho(X, Y) = \frac{Cov[X, Y]}{\sigma_X \sigma_Y} = \frac{E[(X - \mu_X)(Y - \mu_Y)]}{\sigma_X \sigma_Y}, \tag{3}$$

where *Cov*[*X, Y*] represents the covariance of *X* and *Y*.

However, the *linear* correlation coefficient in probability theory is different from and cannot be transformed to the one defined in the deterministic correlated model, because the variances of *X* and *Y* in Eq. (3) must be nonzero and finite. However, in the deterministic correlated model, for any link *l*, when none of its correlated links simultaneously appear on a path, the cost of *l* is fixed/deterministic with a variance of 0.

Stochastic correlated model

In many real-life networks, the link weights are uncertain because of inaccurate Network State Information (NSI) [11, 12]. For instance, Papagiannaki et al. [13] showed that the queuing delay distribution can be approximated by a Weibull distribution. Since the Cumulative Density Function (CDF) of a Weibull distribution is log-concave and the CDFs of many common distributions (e.g., Exponential distribution, Uniform distribution, etc.) are log-concave [14, 15], we make, as in [11, 16], a mild (general) assumption that the link weights follow a log-concave distribution.

We first define the *Correlated Group* (CG):

Definition 1   Given is a network $G(\mathcal{N}, \mathcal{L})$ where $\mathcal{N}$ represents a set of N nodes and $\mathcal{L}$ denotes a set of L links. A Correlated Group (CG) is a subset of links $L_{CG} \subseteq \mathcal{L}$, and $\forall l \in L_{CG}, \exists l' \in L_{CG} \backslash \{l\}$, such that l and l' are correlated ($\rho_{l,l'} \neq 1$).

Accordingly, the Maximum Correlated Group (MCG) is defined as a CG with the maximum number of correlated links. If a link l is uncorrelated/independent with all the other links, then we say $\{l\}$ is a single-element MCG. Suppose there are Ω Maximum Correlated Groups (MCGs), and there are $m_i > 0$ links (denoted as $l_1^i, l_2^i, \ldots, l_{m_i}^i$) in the i-th MCG, where $1 \leq i \leq \Omega$. In the i-th MCG, a multivariate m_i-dimensional log-concave Cumulative Density Function $CDF_i(x_1, x_2, \ldots, x_{m_i})$ is given to allocate cost $x_1, x_2, \ldots, x_{m_i}$ for links $l_1^i, l_2^i, \ldots, l_{m_i}^i$, respectively.

Therefore, if the possible cost of link l ranges from 0 to $w_l^{\max}$ ($0 < w_l^{\max}$), then the probability of allocating a cost value out of this range is 0. Hence, we have $CDF_i(w_l^{\max}) = 1$ for a single-element MCG i, and $CDF_j(w_{l_1^j}^{\max}, w_{l_2^j}^{\max}, \ldots, w_{l_{m_j}^j}^{\max}) = 1$ for a multi-element MCG j.

For example, Fig. 2 shows a 2-dimensional multivariate normal distribution, where both variables are in the range [0, 4] with mean 2 and covariance matrix $\begin{bmatrix} 0.9 & 0.4 \\ 0.4 & 0.3 \end{bmatrix}$. Similarly to Eq. (3), the correlation matrix (composed of linear correlation coefficients) can be derived from the covariance matrix and the variables' standard variances in the multivariate normal distribution. However, we do not explicitly use the *linear* correlation coefficient in the stochastic correlated model, since we will later prove that via the log-concave property of this model, the shortest path problem can be solved by convex optimization.

Yang *et al. Compu Social Networls (2016) 3:1*

Page 5 of 20

Fig. 2 A 2-dimensional multivariate normal distribution
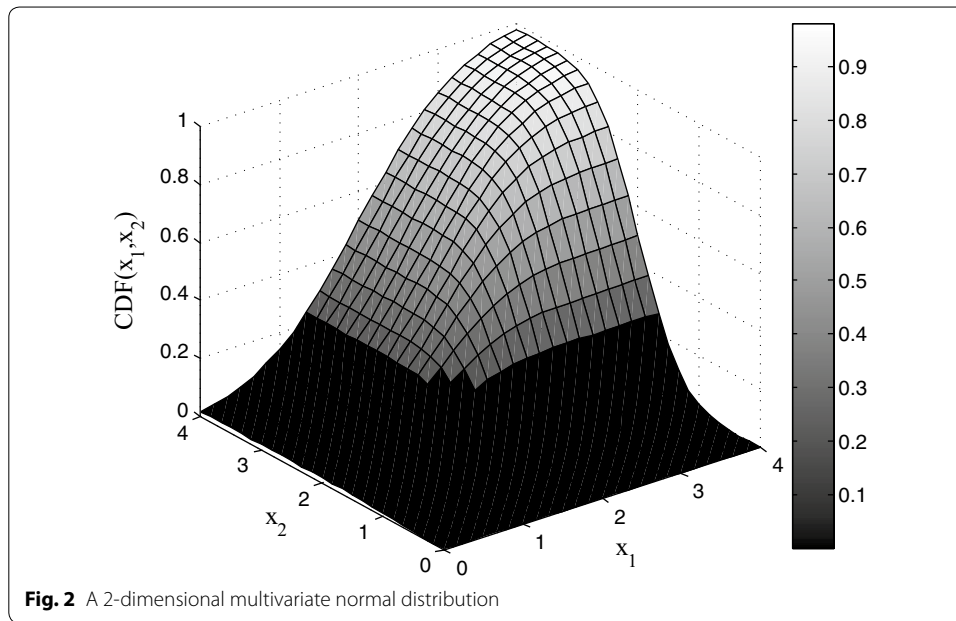
Shortest paths in correlated networks
Shortest path under the deterministic correlated model

Definition 2   Given is a directed network $G(\mathcal{N}, \mathcal{L})$, and each link $l \in \mathcal{L}$ has a cost $w(l)$ following the deterministic correlated model. The Shortest Path under the Deterministic Correlated Model (SPDCM) problem is to find a path from a source s to a destination t with minimum cost.

In conventional deterministic networks, a subpath of a shortest path is also the shortest. We refer to this property as the dominance of the subpath. However, this is not the case in networks with deterministic correlated link weights, which means that a dominated path may also lead to an optimal solution. For instance, in the example of Fig. 1, we can see that although subpath $s-b$ has a smaller cost than subpath $s-a-b$, path $s-a-b-t$ (instead of path $s-b-t$) has minimum cost. In the following, we will study the complexity of the SPDCM problem.

Theorem 1   *The SPDCM problem is NP-hard.*

Proof   When the correlation coefficient is equal to 1, the SPDCM problem can be solved in polynomial time by a conventional shortest path algorithm. We therefore prove in the following that the SPDCM problem is NP-hard for "increasing correlation" >1 as well as "decreasing correlation" <1.

Increasing correlation

When the correlation coefficient is greater than 1, we make a reduction to the forbidden pairs shortest path problem, which is known to be NP-hard [17]. In a given network and for a given set of node pairs ζ, the forbidden pairs shortest path problem looks for the shortest path between s and t such that at most one node from each pair in the set

Yang *et al. Compu Social Networls (2016) 3:1*

Page 6 of 20

ζ lies on this path. Let us consider a network with deterministic correlated link weights. When two nodes i and j form a forbidden pair, their costs are correlated such that $w(i,.) \oplus w(j,.) = \infty$, where $(i,.)$ and $(j,.)$ represent any link that contains an end node of i and j, respectively. In all the other cases, the link costs are uncorrelated and finite. Since $w(i,.) \oplus w(j,.) = \infty$, if the two forbidden nodes appear in the same path, then the cost of this path will be ∞, so it will never lead to the shortest path. Now, the SPDCM problem is equivalent to the forbidden pairs shortest path problem.

Decreasing correlation

When the correlation coefficient is less than 1, we make a reduction to the Minimum Color Single-Path (MCSiP) problem, which is NP-hard [18]. Given a network $G(\mathcal{N}, \mathcal{L})$, and given the set of colors $C = \{c_1, c_2, \ldots, c_g\}$ where g is the total number of colors, and given the color set $\{c_l\}$ associated to each link $l \in \mathcal{L}$, the Minimum Color Single-Path (MCSiP) problem is to find one path from source node s to destination node t such that it uses the least amount of colors.

Assume that each color c_i is associated with cost 1, where $1 \le i \le g$. We further assume that $w(l_1) \oplus w(l_2) \oplus \cdots \oplus w(l_m) = q$, where q is the total number of distinct colors belonging to these m links. Therefore, the SPDCM problem is equivalent to the MCSiP problem. □

Theorem 2 *The SPDCM problem cannot be approximated to arbitrary degree in polynomial time, unless $P = NP$.*

Proof We provide a proof by contradiction.

Increasing correlation

Assume that a polynomial-time approximation algorithm exists that can find a path with a cost at most $\alpha \cdot opt$, where $\alpha > 1$ is an approximation ratio. For a pair of forbidden nodes i and j, we further assume $w(i,.) \oplus w(j,.) > \alpha * opt$. Therefore, if an approximation algorithm can find a path ψ with cost at most $\alpha * opt$ from s to t, then i and j cannot be simultaneously traversed by this path ψ, which means that the forbidden pairs shortest path problem can be solved in polynomial time, which results in a contradiction.

Decreasing correlation

We first introduce the Disjoint Connecting Paths problem [19]. Given a directed network $G(\mathcal{N}, \mathcal{L})$, a collection of disjoint node pairs $(s_1, t_1), (s_2, t_2), \ldots, (s_z, t_z)$, does G contain z mutually link-disjoint paths, one connecting s_i and t_i for each i, $1 \le i \le z$. This problem is NP-hard when $z \ge 2$. Assume that a polynomial-time approximation algorithm exists that can find a path with a cost at most $\alpha \cdot opt$, where $\alpha > 1$ is an approximation ratio. Assuming that all the links in the network have weight 1, and link (u, v) and any $m > 0$ links in $\mathcal{L} \setminus \{(u, v)\}$ are correlated, with a total cost of $\frac{1}{\beta} \cdot m$. Moreover, any two or more links in $\mathcal{L} \setminus \{(u, v)\}$ are assumed to be uncorrelated/independent.

According to this assumption, the minimum value of a shortest path is 1 if link (u, v) is not traversed, i.e., it traverses only one link from s to t. However, the optimal solution which traverses link (u, v) has a total cost of $opt = \frac{1}{\beta} \cdot c$, where c is the sum of minimum hops from s to u and from v to t. For any given α, let $\frac{\beta}{c} > \alpha$, then $1 > \alpha \cdot \frac{1}{\beta} \cdot c$, which means $1 > \alpha \cdot opt$. To find a path with cost at most $\alpha \cdot opt$, the polynomial-time algorithm must find a path which traverses link (u, v). In that case, the algorithm can, in

Yang *et al. Compu Social Networks (2016) 3:1*

Page 7 of 20

polynomial time, find two link-disjoint paths from s to u, and from v to t, which results in a contradiction. □

Next, we study the performance of a conventional shortest path algorithm running on a graph where each link has an "uncorrelated" weight value.

Lemma 1 *When all the correlation coefficients are greater than 1, a conventional shortest path ψ has a total cost at most $\frac{\rho_{\max}}{\rho_{\mathrm{opt}}} \cdot opt$, where $\rho_{\max}$ and ρ_{opt} are the largest correlation coefficient and the optimal solution's correlation coefficient, respectively, and opt is the cost of the optimal solution.*

Proof Let $U(\psi) = \sum_{l \in \psi} w(l)$ and let $C(\psi) = \rho_u \cdot U(\psi) = \rho_u \cdot \sum_{l \in \psi} w(l)$ reflect the total joint cost of path ψ considering their correlation, where ρ_u indicates the correlation coefficient of path ψ. On one hand, a conventional shortest path ψ should satisfy $U(\psi) \leq \frac{\mathrm{opt}}{\rho_{\mathrm{opt}}}$. On the other hand, $C(\psi) \leq \rho_{\max} \cdot U(\psi)$ considering that $\rho_{\max}$ is the largest correlation coefficient. Hence, $C(\psi) \leq \rho_{\max} \cdot U(\psi) \leq \frac{\rho_{\max}}{\rho_{\mathrm{opt}}} \cdot opt$. □

Lemma 2 *When all the correlation coefficients are less than 1, a conventional shortest path ψ has a cost at most $\frac{1}{\rho_{\min}} \cdot opt$, where $\rho_{\min}$ is the smallest correlation coefficient among all the correlation coefficients.*

Proof Let $V(\psi) = \sum_{l \in \psi} w(l)$ and let $C(\psi) = \rho_u \cdot \sum_{l \in \psi} w(l)$ reflect the total joint cost of path ψ considering their correlation. Since all the correlations are decreasing ($\rho < 1$), we have $C(\psi) \leq V(\psi)$. On the other hand, $\rho_{\min} \cdot V(\psi) \leq opt$ considering that $\rho_{\min}$ is the smallest correlation coefficient. Hence, $C(\psi) \leq V(\psi) \leq \frac{1}{\rho_{\min}} \cdot opt$. □

Via Lemmas 1 and 2, we obtain Theorem 3.

Theorem 3 *In a network with links following the deterministic correlated model, a conventional shortest path can have cost at most $\max(\frac{\rho_{\max}}{\rho_{\mathrm{opt}}}, \frac{1}{\rho_{\min}}) \cdot opt$.*

Theorem 3 reveals that a conventional shortest path may have arbitrarily bad performance, since either $\frac{\rho_{\max}}{\rho_{\mathrm{opt}}}$ can be infinitely large or $\rho_{\min}$ can be infinitely small.

An exact algorithm to solve the SPDCM problem

To solve the SPDCM problem exactly, we modify Dijkstra's algorithm by letting each node store as many subpaths as possible, which is similar to the exact algorithm for solving the multi-constrained routing problem [20]. Since each node can store as many subpaths as possible, its running time is exponential. We start with some notations used in the algorithm:

sus[u][h]: the parent node of the h-th subpath from s to u.

dist[u][h]: the cost value of the h-th subpath from s to u.

counter[u]: the number of stored subpaths at node u.

u[m]: the m-th subpath from s to u.

adj(u): the set of nodes adjacent to node u.

The pseudo-code of the exact algorithm is given in Algorithm 1.

Yang *et al. Compu Social Neworls (2016) 3:1*

Page 8 of 20

Algorithm 1 $SPDCM(G, s, t)$

1: $Q \leftarrow s$, $P \leftarrow \emptyset$, $dist[s][1] \leftarrow 0$, $dist[i][h] \leftarrow \infty$, $sus[i][h] \leftarrow i$, $counter[s] \leftarrow 1$, $counter[i] \leftarrow 0$, $\forall i \in \mathcal{N} \backslash \{s\}$.
2: **While** $Q \neq \emptyset$
3:     $u[m] \leftarrow$ Extract-min(Q)
4:     **If** ($u == t$) **do**
5:         Insert (P, $u[m]$)
6:     **Else**
7:         **Foreach** $v \in adj(u)$ **do**
8:             $counter(v) = counter(v) + 1$
9:             Calculate the total cost of subpath $u[m] \rightarrow v$ and assign it to $dist[v][counter(v)]$
10:           $sus[v][counter(v)] \leftarrow u$
11:           Insert ($Q, v, counter(v)$)
12: return min(P).

The time complexity of Algorithm 1 can be computed as follows. Let k_{max} denote the maximum number of subpaths for each node to store, then in Step 2 Q contains at most $k_{max}N$ subpaths. According to [21], $k_{max} \leq \lfloor e(N-2)! \rfloor$, where $e \approx 2.718$. When using a Fibonacci heap to structure the heap, selecting the minimum cost path has a time complexity of $O(\log(k_{max}N))$ [22] in Step 3. Steps 7–11 take at most $O(k_{max})$ time for each link to be iterated and hence result in $O(k_{max}L)$ time; because for a fixed link, the steps within the inner loop (Steps 8–11) all cost $O(1)$ time. Step 12 invokes $O(k_{max})$ time for node t to select the minimum cost path. Hence, the overall time complexity of Algorithm 1 is $O(k_{max}N \log(k_{max}N) + k_{max}L)$.

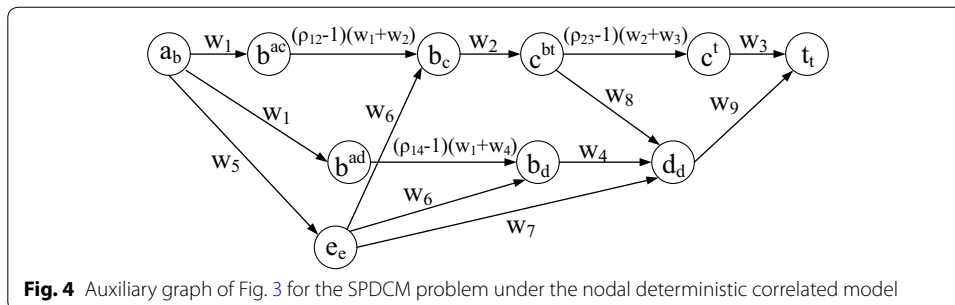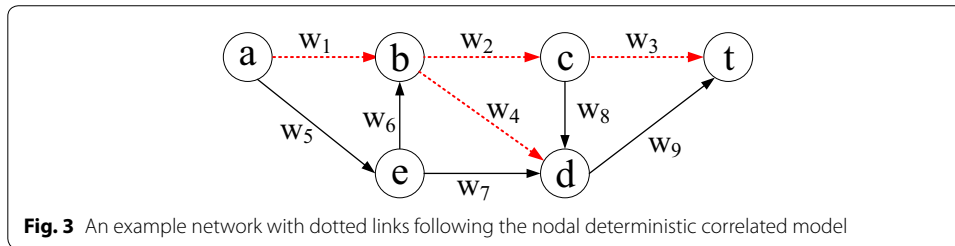**Shortest path under the nodal deterministic correlated model**

In some real-world networks (e.g., SRLG networks), the links that are spatially (geographically) close to each other are usually correlated, whereas the links that are located far from each other are usually uncorrelated. We make an additional assumption, which is that only the links sharing the same node can be correlated. We call this *nodal* correlation.

Although the SPDCM problem is NP-hard, we will show that, by transforming the original graph to an auxiliary graph, the Shortest Path under the Nodal Deterministic Correlated Model (SPNDCM) problem is solvable in polynomial time. For any node a, there are generally two cases of nodal correlation, namely (1) links in the form of (a, b) and (a, c), and (2) links in the form of (a, b) and (b, c) are correlated. When (a, b) and (a, c) are correlated, a simple path cannot traverse both of them, since looping is not allowed. In this sense, any simple path only traverses at most one of them, which means that the links' correlation will not affect the cost calculation of any simple path. Therefore, we only need to consider the case when (a, b) and (b, c) are correlated. We first define that if (a, b) and (b, c) are correlated, then a and b are called *correlated nodes*, which is represented by C_n, else they are *uncorrelated nodes*, which is denoted by U_n. Subsequently, based on the original graph $G(\mathcal{N}, \mathcal{L})$, the auxiliary graph $G^A(\mathcal{N}^A, \mathcal{L}^A)$ can be constructed as follows:

1. For any two links $(u, v) \in \mathcal{L}$ and $(v, y) \in \mathcal{L}$ that are correlated in G, create new nodes u_v, v^{uy}, v_y, and y_y in G^A if they do not already exist. For node $v \in \mathcal{N}$ in $\mathcal{G}$, in case v^{uy} and v_v need to be created, create v^{uy} only and regard v^{uy} to be the same as v_v.
2. For any node $a \in \mathcal{N}$ and if it is an *uncorrelated node* (in U_n), create node a_a in G^A.

3. For any two correlated links (u, v) and (v, y) in G, create links (u_v, v^{uy}), (v^{uy}, v_y), and (v_y, y_y) in $\mathcal{G}^A$. Assign to the links (u_v, v^{uy}) and (v_y, y_y) the weights of $w(u, v)$ and $w(v, y)$, respectively, and the link (v^{uy}, v_y) with weight $(\rho_{(u,v)(v,y)} - 1) \cdot (w(u, v) + w(v, y))$, where $\rho_{(u,v)(v,y)}$ is the correlation coefficient of links (u, v) and (v, y).

4. For each link $(a, b) \in \mathcal{L}$ such that both node a and node b are not *correlated nodes*, create the link (a_a, b_b) also in G^A with the link weight of $w(a, b)$.

5. For each link $(a, b) \in \mathcal{L}$ such that $a \in U_n$ and $b \in C_n$, draw links (a_a, b_r) in G^A, where $r \in \mathcal{N}$ and $b_r \in G^A$.

6. For each link $(a, b) \in \mathcal{L}$ such that $a \in C_n$ and $b \in U_n$, draw links (a^{rz}, b_b) in G^A, where $r, z \in \mathcal{N}$ and $a^{rz} \in \mathcal{N}^A$.

The idea of the auxiliary graph is that if two links (u, v) and (v, y) are correlated, we create four corresponding nodes u_v, v^{uy}, v_y, and y_y and then draw three links $(u_v, v^{uy}), (v^{uy}, v_y)$, and (v_y, y_y). We use (u_v, v^{uy}) and (v_y, y_y) to indicate their uncorrelated values, respectively, if only one of these two links is traversed, and (v^{uy}, v_y) to represent the correlated loss (decreasing correlation) or gain (increasing correlation), respectively, if they are traversed simultaneously. For instance, in Fig. 3 where the link weight is labeled above each link, assuming that links (a, b), (b, c), (c, t), and (b, d) are nodal correlated, then Fig. 4 is its corresponding auxiliary graph with the assigned weight shown on each link. In particular, since (a, b), (b, c) and (a, b), (b, d) may have different correlation coefficients, in Fig. 4 we use (b^{ac}, b_c) and (b^{ad}, b_d) to represent their correlation value. Meanwhile, when there is a link from an *uncorrelated node* to a *correlated node* in the original graph, e.g., (e, b) in Fig. 3, we draw links (e_e, b_c) and (e_e, b_d) (Step 5). When there is a link from a *correlated node* to an *uncorrelated node* in the original graph, e.g., (c, d) in Fig. 3, we draw link (c^{bt}, d_d) (Step 6). Considering that there are at most $N(N - 1)$ nodal links in a graph, the original graph can be transferred to the auxiliary graph in polynomial time.

Fig. 3 An example network with dotted links following the nodal deterministic correlated model

Fig. 4 Auxiliary graph of Fig. 3 for the SPDCM problem under the nodal deterministic correlated model

Consequently, running a shortest path algorithm on the auxiliary graph can return a minimum cost path under the nodal deterministic correlated model. Our auxiliary graph can deal with both decreasing and increasing correlation cases. Considering that $(\rho_{(u,v)(v,y)} - 1) \cdot (w(u,v) + w(v,y)) < 0$ in the auxiliary graph under the decreasing correlation case, and Dijkstra's algorithm cannot handle negative link weights, we could for instance run Bellman–Ford algorithm on the auxiliary graph. No negative loops will exist in the auxiliary graph, since if a path traverses the negative weight link (say (v^{uy}, v_y)), it will also traverse the links in the form of (u_v, v^{uy}) and (v_y, y_y), whose total cost is always positive.

Shortest path under the stochastic correlated model

Definition 3 The Shortest Path under the Stochastic Correlated Model (SPSCM) problem: In a given directed graph $\mathcal{G}(\mathcal{N}, \mathcal{L})$ where the link costs follow the stochastic correlated model, it is assumed that there are in total Ω Maximum Correlated Groups (MCGs). The SPSCM problem is to find a path from source s to destination t such that its total cost is minimized and the probability to realize this value is no less than P_s.

We present a convex optimization formulation to solve the SPSCM problem. Convex optimization problems can usually be solved quickly and accurately with convex optimization solvers [23]. Let us first introduce how to develop a Linear Programming (LP) formulation to solve the shortest path problem in deterministic networks:

Objective

$$\min \sum_{(u,v)\in\mathcal{L}} w(u,v) \cdot y_{uv} \tag{4}$$

Constraints

$$y_{uv} \geq 0 \tag{5}$$

$$\sum_{v\in\mathcal{N}} y_{uv} - \sum_{v\in\mathcal{N}} y_{vu} = \begin{cases} 1 & u = s \\ -1 & u = t \\ 0 & \text{otherwise} \end{cases}, \tag{6}$$

where y_{uv} indicates whether link (u, v) is part of the shortest path. When $y_{uv} = 1$, it indicates that link (u, v) appears on the path, else $y_{uv} = 0$. The objective is to minimize the total cost value of the path. Constraint Eq. (6) accounts for that except for s and t, the number of incoming and outgoing links that are part of the path must be the same. For the source node s, the number of its outgoing links should be 1, and for the destination node t the number of its incoming links should be 1. The dual of the above Linear Program (LP) can be expressed as follows:

Objective

$$\max \; d_t \tag{7}$$

Constraints

$$d_s = 0 \tag{8}$$

$$d_v - d_u \leq w(u,v) \;\; \forall (u,v) \in \mathcal{L}, \tag{9}$$

where d_u is a value between 0 and 1. Similarly, the SPSCM problem can be solved by the following convex formulation:

Objective

$$\max \; d_t \tag{10}$$

Constraints

$$d_s = 0 \tag{11}$$

$$d_v - d_u \leq x(u,v) \;\; \forall (u,v) \in \mathcal{L} \tag{12}$$

$$\sum_{i \in \Omega} -\log \left(CDF_i \left(x(l_1^i), x(l_2^i), \ldots, x(l_{m_i}^i) \right) \right) \leq -\log(P_s), \tag{13}$$

where the variables $x(u,v), x(l_1^i), x(l_2^i), \ldots, x(l_{m_i}^i)$ indicate the allocated possible cost of links (u,v), l_1^i, l_2^i, ..., $l_{m_i}^i$, respectively. Constraint (13) ensures that the total probability of realizing the total cost is no less than P_s. In Eq. (13), for each MCG we apply the multi-dimensional CDF functions to calculate the probability of realizing a cost. Since the multi-dimensional CDF function is log-concave, $-\log \left(CDF_i(x(l_1^i), x(l_2^i), \ldots, x(l_{m_i}^i)) \right)$ is convex, and by summing all the MCGs' CDFs together, it remains convex, which indicates that Eq. (13) is convex. The other constraints are also convex, which proves that the above formulation is a convex optimization formulation.

Widest path under the deterministic correlated model

The Widest Path in Deterministic Networks (WPDN) problem is to find a path from s to t such that the minimum link weight among all its traversed links is maximized. This problem appears with bottleneck metrics, such as bandwidth. The WPDN problem is solvable in polynomial time: First, we order all the link weights in the network in increasing order. After that, each round we prune lowest weight links in the graph and run a Depth First Search (DFS) or a Breadth First Search (BFS) algorithm to find a path from s to t. The algorithm will end if there is no path anymore from s to t and return the pruned weight value of the previous round.

In the Widest Path under the Deterministic Correlated Model (WPDCM) problem, if $m > 1$ correlated links in a path have a joint weight value W, then for each link the maximum average/amortized weight is $\frac{W}{m}$. For instance, if a path traverses three correlated links with a joint weight value of 15 and passes another uncorrelated link with a weight of 6, then this path has a "width" value of 5. The reason is that the maximum (average/amortized) weight for each of these three correlated links is $15/3 = 5$, and this value is less than for another uncorrelated link (6).

However, the WPDCM problem is still NP-hard and cannot be approximated to arbitrary degree. The proof follows analogously from Theorems 1 and 2.

Yang *et al. Compu Social Networls (2016) 3:1*

Page 12 of 20

Minimum cuts in correlated networks

Min-cut under the deterministic correlated model

Definition 4 The Min-Cut under the Deterministic Correlated Model (MCDCM) problem: Given is a network $G(\mathcal{N}, \mathcal{L})$, and each link $l \in \mathcal{L}$ is associated with a cost $w(l)$. It is assumed that two or more link costs are correlated under the deterministic correlated model. Given a source s and a target t, find a cut $\mathcal{C}$ that partitions G into two disjoint subsets X ($X \in \mathcal{N}$) and $\mathcal{N} - X$ such that s and t are in different subsets and the cost of the cut C is minimized.

The min-cut value is not equal to the maximum flow value under the deterministic correlated model. For example, in Fig. 5 assume that links (s, a) and (s, t) are correlated with a joint cost of 8. In this example, the maximum flow from s to t is $s - a - t$ with value 10, while the min-cut is composed of links (s, a) and (s, t) and has a cost of 8.

Theorem 4 *The MCDCM problem is NP-hard.*

Proof In Fig. 6, we assume that the links in the form of (x_i, x_{i+1}) and (z_{i+1}, z_i) have infinite uncorrelated cost and the link costs of (x_i, y_i) and (y_i, z_i) follow the deterministic correlated model, where $1 \leq i \leq n$. We want to find a min-cut to separate x_1 and z_1. Based on Fig. 6, we first derive Fig. 7 with the same nodes except that we add one more node s. We set $s = y_0$, and $t = y_n$. The link weight in Fig. 7 is set as follows: (y_{i-1}, x_i) and (y_{i-1}, z_i) have 0 uncorrelated cost, while (x_i, y_i) and (z_i, y_i) have the same (correlated) costs with (x_i, y_i) and (y_i, z_i) in Fig. 6, respectively, where $1 \leq i \leq n$. In Fig. 7, we want to solve the SPDCM problem from the source s to the destination t.

Since we want to find a min-cut that separates x_1 and z_1, any cut in the form of (x_i, y_i) and (y_i, z_i), where $1 \leq i \leq n$, is not the optimal solution. The reason is that this kind of cut only separates y_i and other nodes, but not x_1 and z_1. Moreover, considering that the links in the form of (x_j, x_{j+1}) or (y_j, y_{j+1}) have infinite costs, they cannot be in the optimal solution. Based on above analysis, any feasible cut C should contain one link of either (x_i, y_i) or (y_i, z_i), for all $1 \leq i \leq n$. We prove in the following that the MCDCM problem in Fig. 6 can be reduced to the SPDCM problem in Fig. 7 in polynomial time.

The SPDCM problem to the MCDCM problem: Considering an optimal solution of the SPDCM problem, denote by R_{SPDCM} the set of links in the
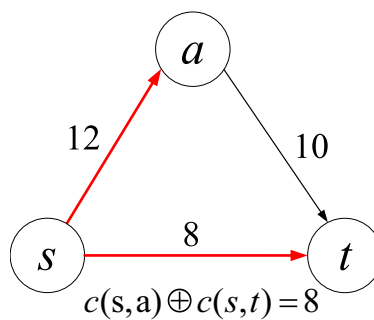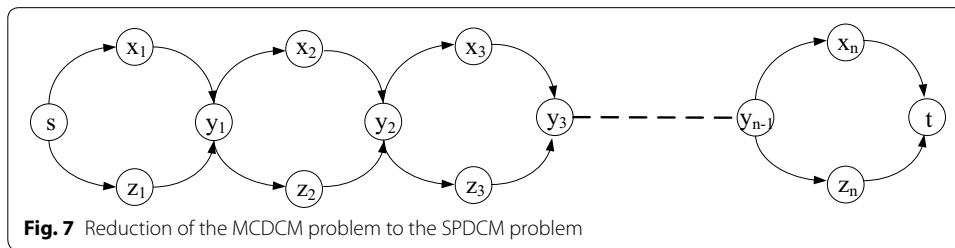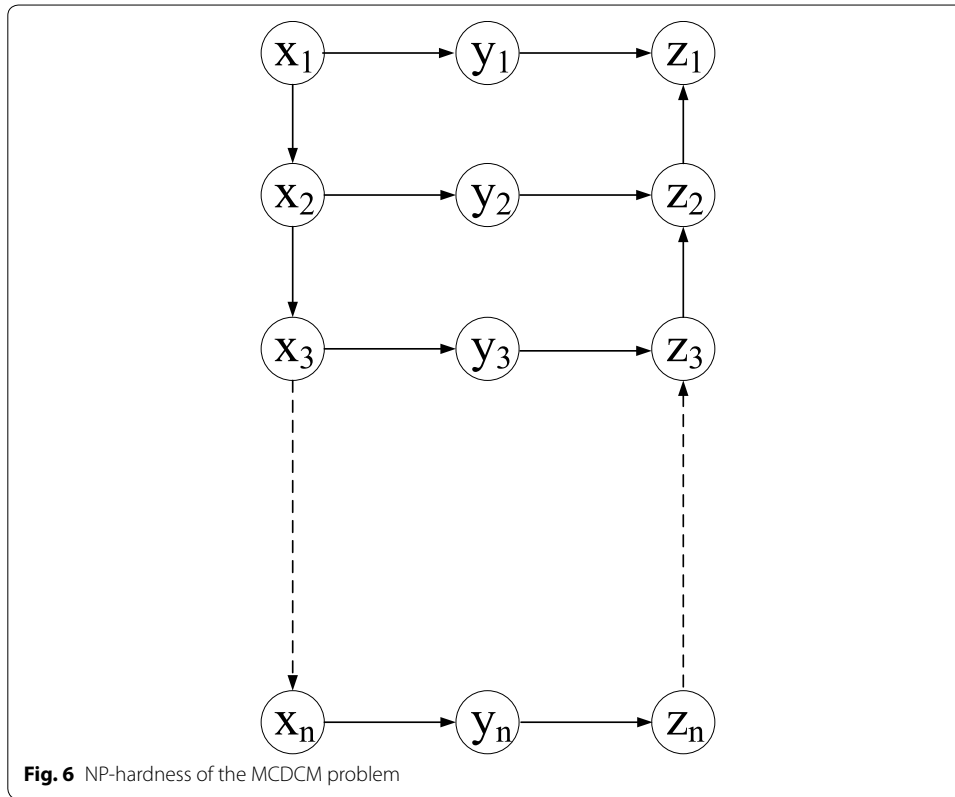

Fig. 5 An example to illustrate that the maximum flow is not equal to the min-cut in correlated networks

Yang *et al. Compu Social Networls (2016) 3:1*

Page 13 of 20

Fig. 6 NP-hardness of the MCDCM problem

Fig. 7 Reduction of the MCDCM problem to the SPDCM problem

optimal solution of the SPDCM problem. Because R_{SPDCM} has minimum cost, let $C_{MCDCM} = R_{SPDCM} \backslash \{(y_i, x_{i+1}), (y_i, z_{i+1})\}$ and then $\forall (z_i, y_i) \in C_{MCDCM}$, change it to (y_i, z_i) in C_{MCDCM}. Since the links (y_i, x_{i+1}) and (y_i, z_{i+1}) have 0 cost, C_{MCDCM} also has minimum cost (the same with R_{SPDCM}). Therefore, solving the SPDCM problem yields a solution to the MCDCM problem.

The MCDCM problem to the SPDCM problem: An optimal solution of the MCDCM problem should be composed of either (x_i, y_i) or (y_i, z_i), where $1 \leq i \leq n$. Denote by C_{MCDCM} the set of links in the optimal solution of the MCDCM problem. Let $R_{SPDCM} = C_{MCDCM}$ and then $\forall (y_i, z_i) \in R_{SPDCM}$, change it to (z_i, y_i) in R_{SPDCM}. Because C_{MCDCM} has minimum cost value and the links in the form of (y_{i-1}, x_i) or (y_{i-1}, z_i) have 0 cost, R_{SPDCM} together with (y_{i-1}, x_i), if $(x_i, y_i) \in R_{SPDCM}$ or (y_{i-1}, z_i) if $(z_i, y_i) \in R_{SPDCM}$, can form a path from *s* to *t* with minimum cost. Hence, a solution to the MCDCM problem can also solve the SPDCM problem. □

Theorem 5 *The MCDCM problem cannot be approximated to arbitrary degree in polynomial time, unless $P = NP$.*

Proof The proof follows from the fact that the SPDCM problem cannot be approximated to arbitrary degree in polynomial time according to Theorem 2.   □

Theorem 6 *By assigning each link l with the cost $w(l)$, running a conventional min-cut algorithm will return a cut with total cost at most $\max(\frac{\rho_{\max}}{\rho_{\text{opt}}}, \frac{1}{\rho_{\min}}) \cdot opt$.*

Proof The proof follows analogously from Theorem 3.   □

Since the MCDCM problem is NP-hard and even does not admit a polynomial-time approximation algorithm, we suggest a brute-force approach to solve it. The idea is that we start with two sets A and B, with s in A and t in B. Then we have $N - 2$ nodes left, and there are $\binom{N-2}{0} + \binom{N-2}{1} + \cdots + \binom{N-2}{N-2} = O(2^N)$ combinations to assign these $N - 2$ nodes to sets A and B. Each combination assignment corresponds to a cut to separate A and B, and the one with minimum cost is returned as the optimal solution.

Min-cut under the SRLG-like correlated model

In "Correlated link weight models" section, we introduced and formulated the joint failure calculation in SRLG networks, which follows the decreasing correlated model. We define the SRLG-like correlated model as follows:

Definition 5 The SRLG-like correlated model: Suppose $l_1, l_2, \ldots, l_m$ $(1 < m \leq L)$ form a Correlated Group (CG), then $w(l_1) \oplus w(l_2) \cdots \oplus w(l_j)$ is greater than the sum of at most $j - 1$ link costs, but smaller than $w(l_1) + w(l_2) \cdots + c(l_j)$, where $1 < j \leq m$.

The Shortest Path under the SRLG-like model (SP-SRLG) problem is NP-hard, since it is a general case of the MCSiP problem introduced in "Shortest paths in correlated networks" section, which is NP-hard [18]. Also the min-cut under the SRLG-like correlated model (MC-SRLG) problem is NP-hard. Similar to the proof that the MCDCM problem is NP-hard in "Minimum cuts in correlated networks" section, the MC-SRLG problem in the form of Fig. 6 can be reduced to the NP-hard SP-SRLG problem.

In the Nodal SRLG-like correlated model, we assumed that only the links that share the same node follow the SRLG-like correlated model. As "Shortest path under the nodal deterministic correlated model" section shows that the Shortest Path under the Nodal Deterministic Correlation Model problem is solvable in polynomial time, we address the min-cut under the Nodal SRLG-like correlated model (MC-NSRLG) problem in the following. In general, the MC-NSRLG problem is still NP-hard. The reason is that for the MC-NSRLG problem in Fig. 8, where the links in the form of (x_1, y_i) are assumed to be correlated, we could derive a graph like in Fig. 7 by duplicating $n - 1$ more x nodes, z nodes, and one y node. We set $s = y_0$ and $t = y_n$. The link weights in Fig. 7 are set as follows: For any two (x, y_i) and (x, y_j) [or (y_i, z_1) and (y_j, z_1)] in Fig. 8, (x_i, y_i) and (x_j, y_j) [or (y_i, z_i) and (y_j, z_j)] in Fig. 7 follow the same correlation. This link weight setting also applies to more than two links. The link weights in the form of (y_{i-1}, x_i) and (y_{i-1}, z_i) have 0 uncorrelated cost, for $1 \leq i < n$. Consequently, based on these link weight

Fig. 8 NP-hardness of the MC-NSRLG problem
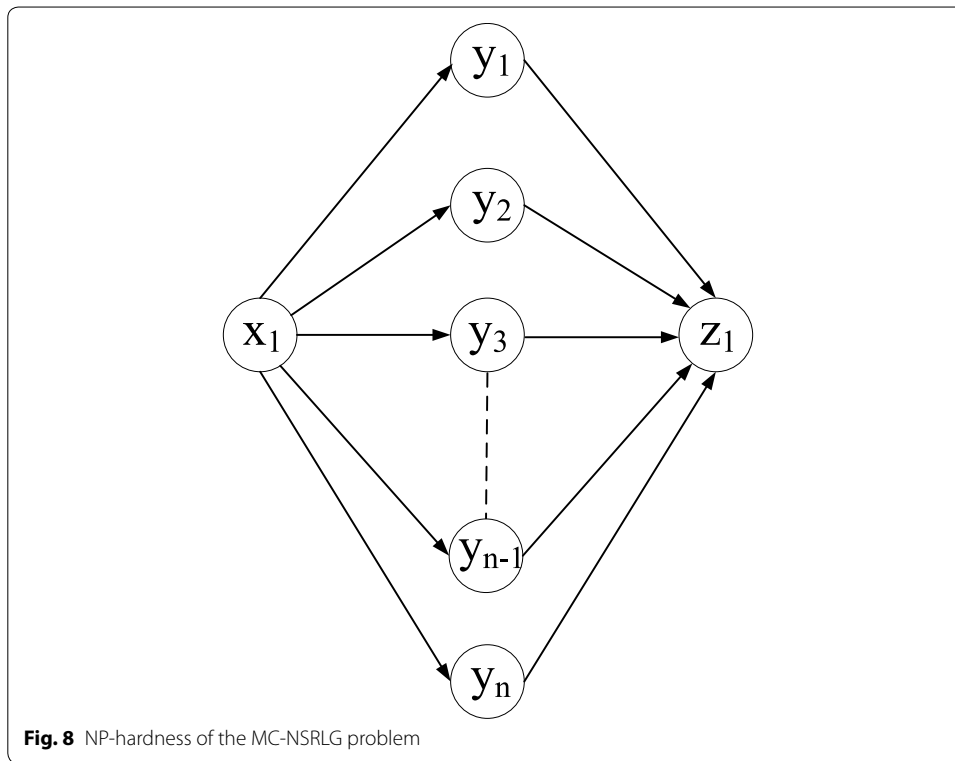
assignments, the MC-NSRLG problem in Fig. 8 can be reduced to the SP-SRLG problem in Fig. 7, similar to the proof of the MCDCM problem in "Minimum cuts in correlated networks" section.

However, we found that the MC-NSRLG problem is solvable in polynomial time when (1) only the nodal links in the form of (u, v) and (v, y) follow the SRLG-like correlated model and/or (2) for any node $u \in \mathcal{N}$, at most two nodal links (u, v) and (u, x) follow the SRLG-like correlated model. To prove case (1), let us first study the following theorem:

Theorem 7 *Any two links in the form of (u, v) and (v, y) will never both appear in the optimal solution of the MC-NSRLG problem in case (1).*

Proof Suppose s and t are separated by a min-cut C such that s is in the subset A and t is in the subset B. A proof by contradiction: we assume that (u, v) and (v, y) are both in the min-cut C. Since C is the min-cut that separates s and t, then node u should be in subset A, otherwise if node u is in subset B, there is no need to use (u, v) and (v, y) as the cut links, since their existence does not affect the connectedness between A and B. Similarly, node v is in subset B, otherwise if node v is also in A, there is no need to cut link (u, v). Based on this analysis, if y is in A, then (v, y) is not necessarily the link in the cut C, since link (v, y) does not affect the connectedness from A to B. However, if node y is in B, link (v, y) is also not necessarily the link in the min-cut, since nodes v and y are in the same subsets, which results in a contradiction.   □

Yang *et al. Compu Social Networls (2016) 3:1*

Page 16 of 20

Based on Theorem 7, we can use conventional Linear Programming (LP) for solving the MC-NSRLG problem under case (1). Following [24], the LP is as follows:

Objective

$$\min \sum_{(u,v) \in \mathcal{L}'} c(u,v) \cdot h_{u,v} \tag{14}$$

Constraints

$$h_{s,t} \geq 1 \tag{15}$$

$$h_{u,v} + h_{v,y} \geq h_{u,y}, \ \forall u, v, y \in \mathcal{N} : u \neq v \neq y \tag{16}$$

$$h_{u,v} \geq 0, \forall u, v \in \mathcal{N} : u \neq v, \tag{17}$$

where $c(u, v)$ stands for the link weight (i.e., capacity) of link (u, v) in the deterministic network and $h_{u,v}$ is an indicator denoting whether (u, v) belongs to the cut.

Under case (2), the MC-NSRLG problem can be solved in polynomial time by running the above LP (Eqs. (14)–(17)) on an auxiliary graph $G^U(\mathcal{N}^U, \mathcal{L}^U)$. The auxiliary graph can be derived from the original graph G as follows:

1. For each pair of two links $(u, v) \in \mathcal{L}$ and $(u, x) \in \mathcal{L}$ that are correlated in G, create a new node u', and draw link (u', u) with weight $\rho_{vx}(w(u, v) + w(u, x))$ to represent the joint cost of links (u, v) and (u, x), where ρ_{vx} represents the correlation coefficient between (u, v) and (u, x).
2. For any link $(a, u) \in \mathcal{L}$ and $(u, b) \in \mathcal{L} \backslash \{v, x\}$ such that (u, v) and (u, x) are correlated in G, draw link (a, u') and (u', b) in G^U with weights $w(a, u)$ and $w(u, b)$, respectively.
3. For each pair of two links $(v, u) \in \mathcal{L}$ and $(x, u) \in \mathcal{L}$ that are correlated in G, create a new node u', and draw link (u, u') with weight $\rho_{vx}(w(v, u) + w(x, u))$ to represent the total cost of links (v, u) and (x, u), where ρ_{vx} represents the correlation coefficient between (v, u) and (x, u).
4. For any link $(a, u) \in \mathcal{L} \backslash \{v, x\}$ and $(u, b) \in \mathcal{L}$ such that (v, u) and (x, u) are correlated in G, draw link (a, u') and (u', b) in G^U with weights $w(a, u)$ and $w(u, b)$, respectively.
5. For the other links $(c, d) \in \mathcal{L}$, create link (c, d) also in G^U with the same weight.

The proposed auxiliary graph shares similarities with the auxiliary graph in Fig. 4. For example, Fig. 10 is an auxiliary graph of the original graph shown in Fig. 9. Moreover, we
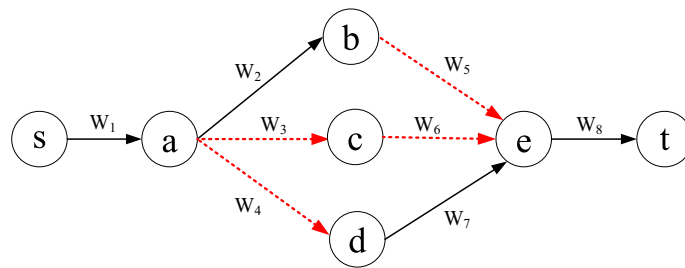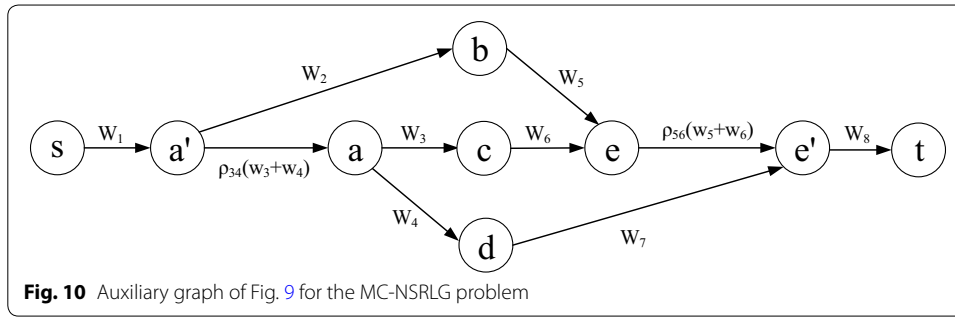

Fig. 9 An example network with dotted links following the SRLG-like correlated model

Fig. 10 Auxiliary graph of Fig. 9 for the MC-NSRLG problem

mention that the proposed auxiliary graph also applies when there are $m > 2$ links starting from the same node that follow the SRLG-like correlated model, but the failure of one correlated link will trigger the other $m - 1$ links to simultaneously "fail" (e.g., SRLG networks, interdependent networks).

Min-cut under the stochastic correlated model

Definition 6   The Min-Cut under Stochastic Correlated Model (MCSCM) problem: In a given directed graph $\mathcal{G}(\mathcal{N}, \mathcal{L})$ with link costs following the stochastic correlated model, the MCSCM problem is to find a cut $\mathcal{C}$ which partitions G into two disjoint subsets X ($X \in \mathcal{N}$) and $\mathcal{N} - X$ such that

- s and t are in different subsets;
- the allocated cost of the cut C is minimum;
- the total probability of realizing the cost value is no less than P_c.

We propose a corresponding convex optimization formulation based on Eqs. (14)–(17):

Objective

$$\min \sum_{(u,v) \in \mathcal{L}} x(u,v) \cdot h_{u,v} \tag{18}$$

Constraints

$$\sum_{i \in \Omega} - \log \left(CDF_i(x(l_1^i), x(l_2^i), \ldots, x(l_{m_i}^i)) \right) \leq - \log(P_c) \tag{19}$$

$$0 \leq x(u,v) \leq c_{(u,v)}^{\max} \quad \forall (u,v) \in \mathcal{L} \tag{20}$$

$$h_{s,t} \geq 1 \tag{21}$$

$$h_{u,v} + h_{v,y} \geq h_{u,y}, \; \forall u, v, y \in \mathcal{N} : u \neq v \neq y \tag{22}$$

$$h_{u,v} \geq 0, \;\; \forall u, v \in \mathcal{N} : u \neq v \tag{23}$$

where $x(u,v), x(l_1^i), x(l_2^i), \ldots, x(l_{m_i}^i)$ indicate the allocated possible cost by links (u, v), $l_1^i, l_2^i, \ldots, l_{m_i}^i$, respectively. In particular, Eq. (19) ensures that the probability of realizing that the min-cut cost is no less than P_c. More specifically, for each MCG we apply the

Yang *et al. Compu Social Networls (2016) 3:1*

Page 18 of 20

multi-dimensional CDF functions to calculate the probability of realizing a cost. Since the CDF function is log-concave, $-\log\left(CDF_i(x(l_1^i), x(l_2^i), \ldots, x(l_{m_i}^i))\right)$ is convex, and by summing all the MCGs together, it remains convex, which indicates that Eq. (19) is convex.

It remains to show that Eq. (18) is convex. In general, the product of two convex functions is not always convex; however, according to [23, pp. 119], one special case is as follows: "If functions *f* and *g* are convex, both nondecreasing (or nonincreasing) and positive (nonnegative) functions on an interval, then $f \cdot g$ is convex." Therefore, for each $(u, v) \in \mathcal{L}$, $x(u, v) \cdot h_{u,v}$ is convex.

Related work

Routing with correlated link weights

In a network with each link having multiple additive link weight metrics (e.g., delay, cost, jitter, etc.), the Quality of Service (QoS) routing problem is to find a path that satisfies a given constraints vector. Kuipers and Van Mieghem [25] study the QoS routing problem under correlated link weights. Another common source of correlation is Shared-Risk Link Groups (SRLGs). Sometimes one SRLG can also be represented by one color, but they share the same meaning in terms of reliability. In this context, Yuan et al. [18] prove that the Minimum Color Single-Path (MCSiP) problem is NP-hard. Yuan et al. also prove that finding two link-disjoint paths with total minimum distinct amount of colors or least amount of coupled/overlapped colors is NP-hard. Lee et al. [26] propose a probabilistic SRLG framework to model correlated link failures and develop an Integer Nonlinear Programming (INLP) formulation to find one unprotected path or two link-disjoint paths with the lowest failure probability.

There is also some literature dealing with correlated routing problems in stochastic networks [27]. For example, in [28] only two possible states are assumed, which are congested and uncongested, and each state corresponds to a cost value. A probability matrix $P_{a,b}^{u,v,y}$, which represents the probability that if (u, v) is in state *a* then (v, y) is in state *b*, is given. Two similar link weight models, called link-based congestion model and node-based congestion model, are proposed in [29]. Based on these models, [28, 29] define and solve the least expected routing problem, which is to find a path from the source to the destination with minimum expected costs. However, in [28, 29] there are only two possible states for each link and only the correlation of the adjacent links is known. We assume a more general (and different) stochastic correlated model, where as long as the links (not necessarily adjacent) are correlated, their joint CDF for allocating costs is known.

Min-cut in conventional networks

The (s, t) min-cut problem refers to partitioning the network into two disjoint subsets, such that nodes *s* and *t* are in different subsets and the total weight of the cut links is minimized. This problem can be solved by finding the maximum flow from *s* to *t* [30]. There is also a lot of work on the min-cut problem with no specified node pairs (s, t). A summary and comparison of polynomial-time algorithms to solve the min-cut problem can be found in [31]. The fastest algorithm to solve the min-cut problem has a time

Yang *et al. Compu Social Networls (2016) 3:1*

Page 19 of 20

complexity of $O(L \log^3 N)$ and was proposed by Karger [32]. Accordingly, the min-cut problem can be tackled by solving at most $N-1$ times the (s, t) min-cut problem.

Constrained maximum flow

As a dual of the min-cut problem, the maximum flow problem in conventional networks is solvable in polynomial time [30]. However, this problem becomes NP-hard if some constraints are imposed on the links. Suppose that negative disjunctive constraints indicate that a certain set of links cannot be used simultaneously for the optimal solution, while positive disjunctive constraints force at least one of a certain set of links to be present in the optimal solution. Pferschy and Schauer [33] prove that the maximum flow problems with both negative and positive disjunctive constraints are NP-hard and do not admit a Polynomial-Time Approximation Scheme (PTAS). For example, the disjunctive constraint corresponds to the correlated link weights, so the maximum flow problem in correlated networks is also NP-hard and does not admit a PTAS. Assuming the link's bandwidth and delay follow a log-concave distribution, Kuipers et al. [16] propose a polynomial-time convex optimization formulation to find the maximum flow in the so-called stochastic networks. When a delay constraint is imposed on each path, the maximum flow problem is NP-hard. To solve it, Kuipers et al. [16] propose an approximation algorithm and a tunable heuristic algorithm.

Conclusions

In this paper, we have studied the shortest path problem and the min-cut problem in correlated networks under two link weight models, namely (1) the deterministic correlated model and (2) the (log-concave) stochastic correlated model. We have proved that these two problems are NP-hard under the deterministic correlated model, and cannot be approximated to arbitrary degree, unless P = NP. Subsequently, we have proposed exact algorithms to solve them. In particular, we have shown that both of them are solvable in polynomial time under a (constrained) nodal deterministic correlated model. For the stochastic correlated model, we have shown that these two problems can be solved by convex optimization.

References
1. Cui W, Stoica I, Katz RH. Backup path allocation based on a correlated link failure probability model in overlay networks. In: Proceedings of IEEE ICNP; 2002. pp. 236–45.
2. Kuipers FA, Dijkstra F. Path selection in multi-layer networks. Comput Commun. 2009;32(1):78–85.
3. Savage S, Collins A, Hoffman E, Snell J, Anderson T. The end-to-end effects of internet path selection. ACM SIGCOMM Comput Commun Rev. 1999;29:289–99.
4. Kostić D, Rodriguez A, Albrecht J, Vahdat A. Bullet: high bandwidth data dissemination using an overlay mesh. ACM SIGOPS Oper Syst Rev. 2003;37:282–97.

5. Kim K, Venkatasubramanian N. Assessing the impact of geographically correlated failures on overlay-based data dissemination. In: Proceedings of IEEE GLOBECOM; 2010. pp. 1–5.
6. Trajanovski S, Kuipers FA, Ilic A, Crowcroft J, Van Mieghem P. Finding critical regions and region-disjoint paths in a network. IEEE/ACM Trans Netw. 2015;23(3):908–21.
7. Kuipers F, Beshir A, Orda A, Van Mieghem P. Impairment-aware path selection and regenerator placement in translucent optical networks. In: Proceedings of the 18th IEEE international conference on network protocols (ICNP); 2010. pp. 11–20.
8. Buldyrev SV, Parshani R, Paul G, Stanley HE, Havlin S. Catastrophic cascade of failures in interdependent networks. Nature. 2010;464(7291):1025–8.
9. Strand J, Chiu AL, Tkach R. Issues for routing in the optical layer. IEEE Commun Mag. 2001;39:81–7.
10. Dantzig G, Fulkerson DR. On the max-flow min-cut theorem of networks. Linear Inequal Relat Syst. 2003;38:225–31.
11. Lorenz DH, Orda A. QoS routing in networks with uncertain parameters. IEEE/ACM Trans Netw. 1998;6(6):768–78.
12. Guérin RA, Orda A. Qos routing in networks with inaccurate information: theory and algorithms. IEEE/ACM Trans Netw. 1999;7(3):350–64.
13. Papagiannaki K, Moon S, Fraleigh C, Thiran P, Tobagi F, Diot C. Analysis of measured single-hop delay from an operational backbone network. Proc IEEE INFOCOM. 2002;2:535–44.
14. Bagnoli M, Bergstrom T. Log-concave probability and its applications. Econ Theory. 2005;26(2):445–69.
15. Mohtashami Borzadaran GR, Mohtashami Borzadaran HA. Log-concavity property for some well-known distributions. Surv Math Appl. 2011;6:203–19.
16. Kuipers FA, Yang S, Trajanovski S, Orda A. Constrained maxmum flow in stochastic networks. In: Proceedings of IEEE ICNP, North Carolina, USA; 2014. pp. 397–408.
17. Gabow HN, Maheshwari SN, Osterweil LJ. On two problems in the generation of program test paths. IEEE Trans Softw Eng. 1976;2(3):227–31.
18. Yuan S, Varma S, Jue JP. Minimum-color path problems for reliability in mesh networks. Proc IEEE INFOCOM. 2005;4:2658–69.
19. Garey MR, Johnson DS. Computers and intractability: a guide to the theory of np-completeness. New York: W. H. Freeman & Co.; 1979.
20. Van Mieghem P, Kuipers FA. Concepts of exact QoS routing algorithms. IEEE/ACM Trans Netw. 2004;12(5):851–64.
21. Van Mieghem P. Paths in the simple random graph and the Waxman graph. Probab Eng Inf Sci. 2001;15(04):535–55.
22. Cormen TH, Stein C, Rivest RL, Leiserson CE. Introduction to algorithms. 2nd ed. Cambridge: MIT Press; 2001.
23. Boyd S, Vandenberghe L. Convex optimization. New York: Cambridge University Press; 2004.
24. Tamir A. Polynomial formulations of min-cut problems. Manuscript. Department of Statistic and Operations Research, Tel Aviv University, Israel; 1994.
25. Kuipers FA, Van Mieghem P. The impact of correlated link weights on QoS routing. Proc IEEE INFOCOM. 2003;2:1425–34.
26. Lee H-W, Modiano E, Lee K. Diverse routing in networks with probabilistic failures. IEEE/ACM Trans Netw. 2010;18(6):1895–907.
27. Yang S, Kuipers FA. Traffic uncertainty models in network planning. IEEE Commun Mag. 2014;52(2):172–7.
28. Waller ST, Ziliaskopoulos AK. On the online shortest path problem with limited arc cost dependencies. Networks. 2002;40(4):216–27.
29. Fan Y, Kalaba R, Moore J. Shortest paths in stochastic networks with correlated link costs. Comput Math Appl. 2005;49(9):1549–64.
30. Ford LR, Fulkerson DR. Maximal flow through a network. Can J Math. 1956;8(3):399–404.
31. Chekuri CS, Goldberg AV, Karger DR, Levine MS, Stein C. Experimental study of minimum cut algorithms. In: Proceedings of the eighth annual ACM-SIAM symposium on discrete algorithms (SODA); 1997. p. 324–33.
32. Karger DR. Minimum cuts in near-linear time. J ACM. 2000;47(1):46–76.
33. Pferschy U, Schauer J. The maximum flow problem with disjunctive constraints. J Comb Optim. 2013;26(1):109–19.