Why did the shape of your network change? (On detecting network anomalies via non-local curvatures)

Bhaskar DasGupta[*] · **Mano Vikash Janardhanan** · **Farzane Yahyanejad**

Abstract Anomaly detection problems (also called *change-point detection* problems) have been studied in data mining, statistics and computer science over the last several decades (mostly in *non-network context*) in applications such as medical condition monitoring, weather change detection and speech recognition. In recent days, however, anomaly detection problems have become increasing more relevant in the context of *network science* since useful insights for many complex systems in biology, finance and social science are often obtained by representing them via networks. Notions of local and non-local curvatures of higher-dimensional geometric shapes and topological spaces play a *fundamental* role in physics and mathematics in characterizing anomalous behaviours of these higher dimensional entities. However, using curvature measures to detect anomalies in networks is not yet very common. To this end, a main goal in this paper to formulate and analyze curvature analysis methods to provide the foundations of systematic approaches to find *critical components* and *detect anomalies* in networks. For this purpose, we use two measures of network curvatures which depend on non-trivial global properties, such as distributions of geodesics and higher-order correlations among nodes, of the given network. Based on these measures, we precisely formulate several computational problems related to anomaly detection in static or dynamic networks, and provide non-trivial computational complexity results for these

[*]Bhaskar DasGupta (corresponding author)
Department of Computer Science, University of Illinois at Chicago, Chicago, IL 60607, USA
Tel.: +312-355-1319
Fax: +312-413-0024
E-mail: bdasgup@uic.edu

Mano Vikash Janardhanan
Department of Mathematics, University of Illinois at Chicago, Chicago, IL 60607, USA
E-mail: manovikashj@gmail.com

Farzaneh Yahyanejad
Department of Computer Science, University of Illinois at Chicago, Chicago, IL 60607, USA
E-mail: farzanehyahyanejad@gmail.com

problems. This paper must *not* be viewed as delivering the final word on appropriateness and suitability of specific curvature measures. Instead, it is our hope that this paper will stimulate and motivate further theoretical or empirical research concerning the exciting interplay between notions of curvatures from network and non-network domains, a *much* desired goal in our opinion.

1 Introduction

Useful insights for many complex systems are often obtained by representing them as networks and analyzing them using graph-theoretic and combinatorial algorithmic tools [1, 27, 60]. In principle, we can classify these networks into *two* major classes:

▷ *Static* networks that model the corresponding system by *one* fixed network. Examples of such networks include biological signal transduction networks *without* node dynamics, and many social networks.
▷ *Dynamic* networks where elementary components of the network (such as nodes or edges) are added and/or removed as the network *evolves* over time. Examples of such networks include biological signal transduction networks *with* node dynamics, causal networks reconstructed from DNA microarray time-series data, biochemical reaction networks and dynamic social networks.

Typically, such networks may have so-called *critical* (*elementary*) components whose presence or absence alters some significant *non-trivial non-local* property[1] of these networks. For example:

▷ For a static network, there is a rich history in finding various types of critical components dating back to quantifications of fault-tolerance or redundancy in electronic circuits or routing networks. Recent examples of practical application of determining critical and non-critical components in the context of systems biology include quantifying redundancies in biological networks [5, 54, 70] and confirming the existence of central influential neighborhoods in biological networks [2].

---

[1] A non-trivial property usually refers to a property such that a significant percentage of all possible networks satisfies the property and also a significant percentage of all possible networks does *not* satisfy the property. A non-local property (also called global property) usually refers to a property that *cannot* be inferred by simply looking at a *local* neighborhood of any one node.

▷ For a dynamic network, critical components may correspond to a set of nodes or edges whose addition and/or removal *between* two time steps alters a significant topological property (*e.g.*, connectivity, average degree) of the network. Popularly also known as the *anomaly detection* or *change-point detection* [7, 51] problem, these types of problems have also been studied over the last several decades in data mining, statistics and computer science *mostly in the "non-network" context of time series data* with applications to areas such as medical condition monitoring [14, 80], weather change detection [31, 63] and speech recognition [21, 66].

In this paper we seek to address research questions of the following *generic* nature:

"Given a static or dynamic network, identify the critical components of the network that "encode" significant non-trivial global properties of the network".

To identify critical components, one first needs to provide details for following four specific items:

(*i*) network model selection,
(*ii*) network evolution rule for dynamic networks,
(*iii*) definition of elementary critical components, and
(*iv*) network property selection (*i.e.*, the global properties of the network to be investigated).

The specific details for these items for this paper are as follows:

(*i*) Network model selection: Our network model will be undirected graphs.
(*ii*) Network evolution rule for dynamic networks: Our dynamic networks follow the time series model and are given as a sequence of networks over *discrete* time steps, where each network is obtained from the previous one in the sequence by adding and/or deleting some nodes and/or edges.
(*iii*) Critical component definition: Individual edges are elementary members of critical components.
(*iv*) Network property selection: The network measure for this paper will be based on one or more well-justified notions of "network curvature". More specifically, we will use **(*a*) Gromov-hyperbolic curvature** based on the properties of *exact and approximate* geodesics distributions and higher-order connectivities and **(*b*) geometric curvatures** based on identifying network motifs with *geometric complexes* ("geometric motifs" in systems biology jargon) and then using Forman's combinatorializations.

1.1 Organization of the paper and a summary of our contributions

The rest of the paper is organized as described below.

In Section 2 we introduce some basic definitions and notations and provide a summary list of some other notations that are are used throughout the rest of the paper.

In Section 3 we discuss the relevant background, motivation, and justification for using the curvature measures and provide two illustrative examples in which curvature measures detect anomaly where other simpler measures do not. We also remark on the limitations of our theoretical results that may be useful to future researchers.

In Section 4 we define and motivate the two notions of graph curvature that is used in this paper in the following manner:

▷ The Gromov-hyperbolic curvature is introduced in Section 4.1 together with justifications for using them, relevant known results and some clarifying remarks about them.
▷ Generic notions of geometric curvatures are introduced in Section 4.2 together with relevant topological concepts necessary to define them and justifications for using them. The precise definition of the geometric curvature used in this paper is given by Equation (1) in Section 4.2.2.

In Section 5 we present our formalizations of anomaly detection problems on networks based on curvature measures. We distinguish two types of anomaly detection problems in the following manner:

▷ In Section 5.1 we formalize the *Extremal Anomaly Detection Problem* (Problem $\textsc{Eadp}_{\mathfrak{C}}(G, \widetilde{E}, \gamma)$) for "static networks" that do *not* change over time.
▷ In Section 5.2 we formalize the *Targeted Anomaly Detection Problem* ($\textsc{Tadp}_{\mathfrak{C}}(G_1, G_2)$) for "dynamic networks" that *do* change over time.

In Section 6 we present our results regarding the computational complexity of extremal anomaly detection problems for the two types of curvatures in the following manner:

▷ Theorem 1 in Section 6.1 states the computational complexity results for geometric curvatures. Some relevant comments regarding Theorem 1 and an informal overview of its proof techniques appear in Section 6.1.1, whereas the precise technical proofs for Theorem 1 are presented separately in Section 6.1.2.
▷ Theorem 2 in Section 6.2 states the computational complexity results for Gromov-hyperbolic curvature. An informal overview of the proof techniques for Theorem 2 appears in the very beginning of Section 6.2.1, whereas the precise technical proofs for Theorem 2 are presented in the remaining part of the same section.

In Section 7 we present our results regarding the computational complexity of targeted anomaly detection problems for the two types of curvatures in the following manner:

▷ Theorem 3 in Section 7.1 states the computational complexity results for geometric curvatures. An informal overview of the proof techniques for

Theorem 3 appears in Section 7.1.1, whereas the precise technical proofs for Theorem 3 are presented separately in Section 7.1.2.

▷ Theorem 4 in Section 7.2 states the computational complexity results for Gromov-hyperbolic curvature. Some relevant comments regarding Theorem 4 and an informal overview of its proof techniques appear in Section 7.2.1, whereas the precise technical proofs for Theorem 4 are presented separately in Section 7.2.2.

Finally, we conclude in Section 8 with a few interesting research problems for future research.

Remarks on the organization of our proofs

Many of our proofs in Sections 6–7 are long, are complicated or involve tedious calculations. For easier understanding and to make the paper more readable, when appropriate we have included a subsection generically titled "Proof techniques and relevant comments regarding Theorem ......" before providing the actual detailed proofs. The reader *is* cautioned however that these brief subsections are meant to provide some general idea and subtle points behind the proofs and should *not* be considered as a substitution for more formal proofs.

2 Basic definitions and notations

For an undirected unweighted graph $G = (V, E)$ of n nodes $v_1, \ldots, v_n$, the following notations related to G are used throughout:

▶ $v_{i_1} \leftrightarrow v_{i_2} \leftrightarrow v_{i_3} \leftrightarrow \cdots \leftrightarrow v_{i_{k-1}} \leftrightarrow v_{i_k}$ denotes a path of *length* $k - 1$ consisting of the edges $\{v_{i_1}, v_{i_2}\}, \{v_{i_2}, v_{i_3}\}, \ldots, \{v_{i_{k-1}}, v_{i_k}\}$.

▶ $\overline{u, v}$ and $\text{dist}_G(u, v)$ denote a *shortest path* and the distance (*i.e.*, number of edges in $\overline{u, v}$) between nodes u and v, respectively.

▶ $\text{diam}(G) = \max_{v_i, v_j} \{\text{dist}_G(v_i, v_j)\}$ denotes the *diameter* of G.

▶ $G \setminus E'$ denotes the graph obtained from G by removing the edges in E' from E.

A ε-approximate solution (or simply an ε-approximation) of a minimization (*resp.*, maximization) problem is a solution with an objective value no larger than (*resp.*, no smaller than) ε times (*resp.*, $1/\varepsilon$ times) the value of the optimum; an algorithm of *performance* or *approximation ratio* ε produces an ε-approximate solution. A problem is ε-*inapproximable* under a certain complexity-theoretic assumption means that the problem does not admit a polynomial-time ε-approximation algorithm assuming that the complexity-theoretic assumption is true. We will also use other *standard* definitions from structural complexity theory as readily available in any graduate level textbook on algorithms such as [72].

Other specialized notations used in the paper are defined when they are first needed. For the benefit of the reader, we provide a list of some such commonly used notations in the paper with brief comments about them in Table 1. Please see the referring section for exact descriptions of these notations.

Nomenclature or brief explanation	Notation	Referring section
path of length $k-1$	$v_{i_1} \leftrightarrow \cdots \leftrightarrow v_{i_k}$	Section 2
shortest path, distance between nodes u and v	$\overline{u,v}$, $\mathrm{dist}_G(u,v)$	Section 2
diameter of graph G	$\mathsf{diam}(G)$	Section 2
the graph $(V, E \setminus E')$ where $G = (V, E)$	$G \setminus E'$	Section 2
curvature of graph G	$\mathfrak{C}$ or $\mathfrak{C}(G)$	Section 4
geodesic triangle	$\Delta_{u,v,w}$	Section 4.1
Gromov-hyperbolic curvature of $\Delta_{u,v,w}$	$\mathfrak{C}_{\mathrm{Gromov}}(\Delta_{u,v,w})$	Section 4.1
Gromov-hyperbolic curvature of graph G	$\mathfrak{C}_{\mathrm{Gromov}}$ or $\mathfrak{C}_{\mathrm{Gromov}}(G)$	Section 4.1
k-simplex of $k+1$ affinely independent points	$\mathcal{S}(x_0, \ldots, x_k)$	Section 4.2.1
order d association of p-face f^p of a q-simplex	f_d^p	Section 4.2.2
geometric curvature of graph G	$\mathfrak{C}_d^p$ or $\mathfrak{C}_d^p(G)$	Section 4.2.2
Extremal Anomaly Detection Problem, value of its optimal solution	$\mathrm{EADP}_{\mathfrak{C}}(G, \widetilde{E}, \gamma)$, $\mathsf{OPT}_{\mathrm{EADP}_{\mathfrak{C}}}(G, \widetilde{E}, \gamma)$	Section 5.1
Targeted Anomaly Detection Problem, value of its optimal solution	$\mathrm{TADP}_{\mathfrak{C}}(G_1, G_2)$, $\mathsf{OPT}_{\mathrm{TADP}_{\mathfrak{C}}}(G_1, G_2)$	Section 5.2
densest-k-subgraph problem	$\mathrm{D}k\mathrm{S}_3$	Section 6.1.2
minimum node cover problem, cardinality of its optimal solution	MNC, $\mathsf{OPT}_{\mathrm{MNC}}$	Section 7.1.2
triangle deletion problem, cardinality of its optimal solution	TDP, $\mathsf{OPT}_{\mathrm{TDP}}$	Section 7.1.2
Hamiltonian path problem for cubic graphs	$\mathrm{CUBIC\text{-}HP}$	Section 7.2.1

Table 1 A list of some frequently used notations with brief explanations.

3 Background, motivation, justification and illustrative examples

The main purpose of this section is to (somewhat informally) explain to the reader the appropriateness of our curvature measures both from a theoretical and an empirical point of view. We also provide brief comments on the limitations of our theoretical results which may be of use to future researchers.

3.1 Justifications for using network curvature measures

Prior researchers have proposed and evaluated a number of established network measures such as *degree-based measures* (*e.g.*, degree distribution), *connectivity-based measures* (*e.g.*, clustering coefficient), *geodesic-based measures* (*e.g.*, betweenness centrality) and other more novel network measures [5, 9, 22, 55] for analyzing networks. The network measures considered in this paper are "appropriate notions" of *network curvatures*. As demonstrated in published research works such as [2, 68, 75, 76], these network curvature measures saliently

encode *non-trivial higher-order correlation* among nodes and edges that *cannot* be obtained by other popular network measures. Some important characteristics of these curvature measures that we consider are [2, Section (III)][50]:

▶ These curvature measures depend on *non-trivial global* network properties, as opposed to measures such as *degree distributions* or *clustering coefficients* that are *local* in nature or *dense subgraphs* that use *only* pairwise correlations.

▶ These curvature measures can mostly be computed efficiently in polynomial time, as opposed to NP-complete measures such as *cliques* [37], *densest-k-subgraphs* [37], or some types of community decompositions such as *modularity maximization* [25].

▶ When applied to real-world networks, these curvature measures can explain many phenomena one frequently encounters in real network applications that are *not* easily explained by other measures such as:

    ▶ paths mediating up- or down-regulation of a target node starting from the same regulator node in *biological regulatory networks* often have many small crosstalk paths, and

    ▶ existence of congestions in a node that is not a hub in *traffic networks*.

Further details about the suitability of our curvature measures for real biological or social networks are provided in Section 4.1.1 for Gromov-hyperbolic curvature and at the end of Section 4.2.2 for geometric curvatures.

Curvatures are very natural measures of anomaly of higher dimensional objects in mainstream physics and mathematics [11, 15]. However, networks are *discrete objects* that do *not* necessarily have an associated natural geometric embedding. Our paper seeks to adapt the definition of curvature from the non-network domains (*e.g.*, from continuous metric spaces or from higher-dimensional geometric objects) in a suitable way for detecting network anomalies. For example, in networks with sufficiently small Gromov-hyperbolicity and sufficiently large diameter a suitably small subset of nodes or edges can be removed to stretch the geodesics between two distinct parts of the network by an exponential amount. Curiously this kind of property can be shown to have extreme implications on the expansion properties of such networks [10, 26], akin to the characterization of singularities (an extreme anomaly) by geodesic incompleteness (*i.e.*, stretching all geodesics passing through the region infinitely) [42].

3.2 Justifications for investigating the edge-deletion model

In this paper we add or delete edges from a network while keeping the node set the same. This scenario captures a wide variety of applications such as inducing desired outcomes in disease-related biological networks via gene knockout [67, 82], inference of minimal biological networks from indirect experimental evidences or gene perturbation data [3, 4, 73], and finding influential nodes

in social and biological networks [5], to name a few. However, the node addition/deletion model or a mixture of node/edge addition/deletion model is also significant in many applications; we leave investigations of these models as future research topics.

3.3 Two illustrative examples

It is obviously practically impossible to compare our curvatures measures for anomaly detection with respect to *every possible* other network measure that has been used in prior research works. However, we do still provide two illustrative examples of comparing our curvature measures to the well-known *densest subgraph measure*. The densest subgraph measure is defined as follows.

Definition 1 (Densest subgraph measure) Given a graph $G = (V, E)$, the densest subgraph measure find a subgraph (S, E_S) induced by a subset of nodes $\emptyset \subset S \subseteq V$ that maximizes the ratio (density) $\rho(S) \stackrel{\text{def}}{=} \frac{|E_S|}{|S|}$. Let $\rho(G) \stackrel{\text{def}}{=} \max_{\emptyset \subset S \subseteq V} \{\rho(S)\}$ denote the density of a densest subgraph of G.

An efficient polynomial time algorithm to compute $\rho(G)$ using a max-flow technique was first provided by Goldberg [40]. We urge the readers to review the definitions of the relevant curvature measures (in Section 4) and the anomaly detection problems (in Section 5) in case of any confusion regarding the examples we provide.

Extremal anomaly detection for a static network

Consider the extremal anomaly detection problem (Problem EADP in Section 5.1) for a network $G = (V, E)$ of 10 nodes and 20 edges as shown in Fig. 1 using the geometric curvature $\mathfrak{C}_3^2$ as defined by Equation (1). It can be easily verified that $\mathfrak{C}_3^2(G) = 6$ and $\rho(G) = 9/4$. Let $\widetilde{E} = E$ and suppose that we set our targeted decrease of the curvature or density value to be 75% of the original value, *i.e.*, we set $\gamma = 3/4 \times \mathfrak{C}_3^2(G) = 9/2$ for the geometric curvature measure and $\gamma = 3/4 \times \rho(G) = 27/16$ for the densest subgraph measure. It is easily verified that $\mathfrak{C}_3^2(G \setminus \{e_1\}) = 1 < 9/2$, thus showing $\mathsf{OPT}_{\text{EADP}_{\mathfrak{C}_3^2}}(G, \widetilde{E}, \gamma) = 1$. However, one can verify that more than 4 edges will need to be deleted from G to bring down the value of $\rho(G)$ to $27/16$ in the following manner: since the densest subgraph in G is induced by 8 nodes and 18 edges, if no more than 4 edges are deleted then the density of this subgraph in the new graph is at least $14/8 > 27/16$.

Targeted anomaly detection for a dynamic biological network

Consider the targeted anomaly detection problem (Problem TADP in Section 5.2) using the Gromov-hyperbolic curvature (Definition 2). Suppose that we have a biological dynamical system of 4 variables x_1, x_2, x_3, x_4 generated

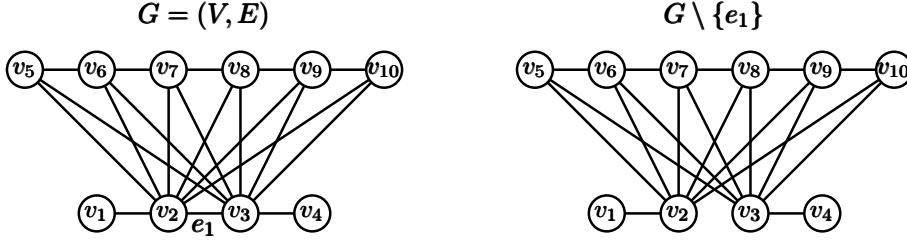$$G = (V, E) \qquad\qquad\qquad\qquad G \setminus \{e_1\}$$

Fig. 1 Toy example of extremal anomaly detection discussed in Section 3.3. The given graph $G = (V, E)$ has $|V| = 10$ nodes, $|E| = 20$ edges and $\alpha = 16$ triangles (3-cycles) giving $\mathfrak{C}_3^2(G) = |V| - |E| + \alpha = 6$, where the densest subgraph of G is the subgraph node-induced by the nodes $V \setminus \{v_1, v_4\}$ with 8 nodes and 18 edges giving $\rho(G) = 18/8 = 9/4$. The graph $G \setminus \{e_1\}$ has $|E \setminus \{e_1\}| = 19$ edges and $\alpha' = 10$ triangles giving $\mathfrak{C}_3^2(G \setminus \{e_1\}) = |V| - |E \setminus \{e_1\}| + \alpha' = 1$. However, it can be verified that more than four edges will need to be deleted from G to bring down the value of $\rho(G)$ to at most $27/16$.
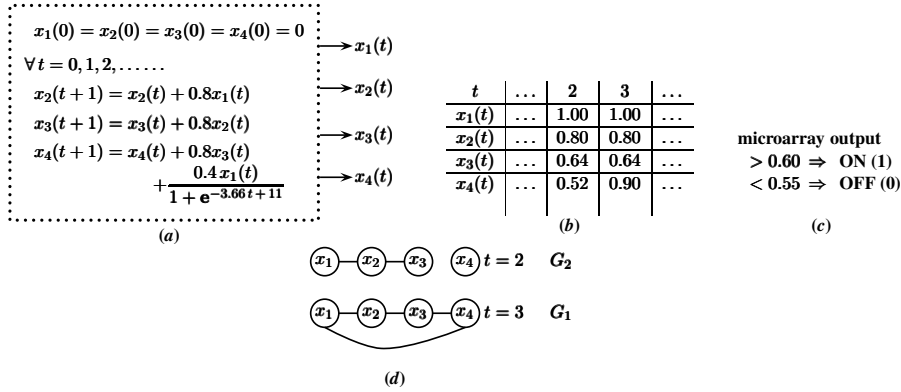

Fig. 2 Toy example of targeted anomaly detection discussed in Section 3.3. (**a**) The original dynamical system with four observable output variables. (**b**) Values of the four output variable over time starting with an all-zero initial condition. (**c**) The thresholding rules employed to binarize the expression values in (**b**) as captured by a DNA microarray. (**d**) The reverse-engineered network for two successive time steps.

by a set of recurrence equations as shown in Fig. 2 (*a*) for $x_1(t), x_2(t), x_3(t)$ and $x_4(t)$ as a function of discrete time $t = 0, 1, 2, 3, \ldots\ldots$, with the initial condition of $x_1(0) = x_2(0) = x_3(0) = x_4(0) = 0$. Note that in this biological system any change in the value of x_1 affects x_4 with a delay. These recurrence equations are not known to the observer, but they generate a sequence of real values of the state variables for each successive discrete time units (shown in Fig. 2 (*b*) for $t = 2$ and $t = 3$). Suppose that an observer measures a binarized version of these real values of the state variables for each successive discrete time units using a DNA microarray by using thresholds as shown in Fig. 2 (*c*), and then reverse-engineers a time-varying network by using the hitting-set approach of Krupa [27, Section 5.4.2][46] with a time delay of 2 (the corresponding network for $t = 2$ and $t = 3$ is shown in Fig. 2 (*d*)). Suppose that

for our targeted anomaly detection problem we fix our attention to the two graphs G_2 and G_1 constructed in the two successive time steps $t = 2$ and $t = 3$, respectively, where $G_2 = G_1 \setminus \{\{x_1, x_4\}, \{x_3, x_4\}\}$ is the target graph. It can be easily verified that $\mathfrak{C}_{\mathrm{Gromov}}(G_1) = \rho(G_1) = 1$, $\mathfrak{C}_{\mathrm{Gromov}}(G_2) = 0$, and $\rho(G_2) = {}^1\!/_2$. Since $\mathfrak{C}_{\mathrm{Gromov}}(G_1 \setminus \{x_1, x_4\}) = 0$ it follows that we only need to delete the edge $\{x_1, x_4\}$ to bring down the value of $\mathfrak{C}_{\mathrm{Gromov}}(G_1)$ to $\mathfrak{C}_{\mathrm{Gromov}}(G_2)$. However, both the edges $\{x_1, x_4\}$ and $\{x_3, x_4\}$ need to be deleted from G_1 to bring down the value of $\rho(G_1)$ to $\rho(G_2)$.

3.4 Brief remarks regarding the limitations of our theoretical results

Our theoretical results obviously have some limitations, specially for real-world networks. For example, our inapproximability results for the Gromov-hyperbolic curvature require a high average node degree. Thus, for real-world networks such as scale-free networks the inapproximability bounds may not apply. On another note, for geometric curvatures we only considered the first-order non-trivial measure $\mathfrak{C}_d^2$, but perhaps more salient non-trivial topological properties could be captured by using $\mathfrak{C}_d^p$ for $p > 2$.

4 Two notions of graph curvature

For this paper, a *curvature* for a graph G is a function $\mathfrak{C} \stackrel{\mathrm{def}}{=} \mathfrak{C}(G) : G \mapsto \mathbb{R}$. There are several ways in which network curvature can be defined depending on the type of global properties the measure is desired to affect; in this paper we consider two such definitions as described subsequently.

4.1 Gromov-hyperbolic curvature

This measure for a metric space was first suggested by Gromov in a group theoretic context [41]. The measure was first defined for *infinite* continuous metric space [15], but was later also adopted for *finite* graphs. Usually the measure is defined via *geodesic triangles* as stated in Definition 2. For this definition, it would be useful to consider the given graph G as a metric graph, *i.e.*, we identify (by an isometry) any edge $\{u, v\} \in E$ with the real interval $[0, 1]$ and thus any point in the interior of the edge $\{u, v\}$ can also be thought as a (virtual) node of G. Define a geodesic triangle $\Delta_{u,v,w}$ to be an ordered triple of three shortest paths ($\overline{u,v}$, $\overline{u,w}$ and $\overline{v,w}$) for the three nodes u, v, w in G.

Definition 2 (Gromov-hyperbolic curvature measure via geodesic triangles) For a geodesic triangle $\Delta_{u,v,w}$, let $\mathfrak{C}_{\mathrm{Gromov}}(\Delta_{u,v,w})$ be the minimum number such that $\overline{u,v}$ lies in a $\mathfrak{C}_{\mathrm{Gromov}}(\Delta_{u,v,w})$-neighborhood of $\overline{u,w} \cup \overline{v,w}$, i.e., for every node x on $\overline{u,v}$, there exists a node y on $\overline{u,w}$ or $\overline{v,w}$

such that $\mathrm{dist}_G(x, y) \leq \mathfrak{C}_{\mathrm{Gromov}}(\Delta_{u,v,w})$. Then the graph G has a Gromov-hyperbolic curvature (or Gromov hyperbolicity) of $\mathfrak{C}_{\mathrm{Gromov}} \overset{\mathrm{def}}{=} \mathfrak{C}_{\mathrm{Gromov}}(G)$ where $\mathfrak{C}_{\mathrm{Gromov}}(G) = \min\limits_{u,v,w \in V} \{\mathfrak{C}_{\mathrm{Gromov}}(\Delta_{u,v,w})\}$.

An infinite collection $\mathcal{G}$ of graphs belongs to the class of $\mathfrak{C}_{\mathrm{Gromov}}$-Gromov-hyperbolic graphs if and only if any graph $G \in \mathcal{G}$ has a Gromov-hyperbolic curvature of $\mathfrak{C}_{\mathrm{Gromov}}$. Informally, any infinite metric space has a finite value of $\mathfrak{C}_{\mathrm{Gromov}}$ if it behaves metrically in the large scale as a *negatively curved* Riemannian manifold, and thus the value of $\mathfrak{C}_{\mathrm{Gromov}}$ can be related to the other standard curvatures of a hyperbolic manifold. For example, a simply connected complete Riemannian manifold whose sectional curvature is below $\alpha < 0$ has a value of $\mathfrak{C}_{\mathrm{Gromov}} = O\left(\sqrt{-\alpha}\right)$ (see [65]). This is a major justification of using $\mathfrak{C}_{\mathrm{Gromov}}$ as a notion of curvature of any metric space.

Let ω be the value such that two $n \times n$ matrices can be multiplied in $O(n^\omega)$ time; the smallest current value of ω is about 2.373 [77]. Then the following results computational complexity results are known for computing $\mathfrak{C}_{\mathrm{Gromov}}(G)$ for an n-node graph G.

▷ $\mathfrak{C}_{\mathrm{Gromov}}(G)$ can be *exactly* computed in $O\left(n^{\frac{5+\omega}{2}}\right) = O\left(n^{3.687}\right)$ time [35].
▷ $(1+\varepsilon)$-approximation of $\mathfrak{C}_{\mathrm{Gromov}}(G)$ can be computed in $\tilde{O}\left(\frac{1}{\varepsilon}n^{1+\omega}\right) = \tilde{O}\left(\frac{1}{\varepsilon}n^{3.373}\right)$ time [28], and $(2+\varepsilon)$-approximation of $\mathfrak{C}_{\mathrm{Gromov}}(G)$ can be computed in $\tilde{O}\left(\frac{1}{\varepsilon}n^\omega\right) = \tilde{O}\left(\frac{1}{\varepsilon}n^{2.373}\right)$ time [28][2].
▷ 8-approximation of $\mathfrak{C}_{\mathrm{Gromov}}(G)$ can be computed in $O\left(n^2\right)$ time [16].

It is easy to see that if G is a tree then $\mathfrak{C}_{\mathrm{Gromov}}(G) = 0$. Other examples of graph classes for which $\mathfrak{C}_{\mathrm{Gromov}}(G)$ is a small constant include *chordal graphs*, *cactus of cliques*, *AT-free* graphs, *link graphs of simple polygons*, and *any* class of graphs with a *fixed* diameter. A small value of Gromov-hyperbolicity is often crucial for algorithmic designs; for example, several routing-related problems or the diameter estimation problem become easier for networks with small $\mathfrak{C}_{\mathrm{Gromov}}$ values [17–19, 39]. There are many well-known measures of curvature of a continuous surface or other similar spaces (*e.g.*, curvature of a manifold) that are widely used in many branches of physics and mathematics. It is possible to relate Gromov-hyperbolic curvature to such other curvature notions indirectly via its scaled version, *e.g.*, see [48, 49, 59].

4.1.1 Gromov-hyperbolic curvature and real-world networks

Recently, there has been a surge of empirical works measuring and analyzing the Gromov curvature $\mathfrak{C}_{\mathrm{Gromov}}$ of networks, and many real-world networks (*e.g.*, preferential attachment networks, networks of high power transceivers in a wireless sensor network, communication networks at the IP layer and at other levels) were observed to have a small constant value of $\mathfrak{C}_{\mathrm{Gromov}}$ [8, 47,

---

[2] $\tilde{O}(\cdot)$ is a standard computational complexity notation that omits poly-logarithmic factors.

49, 59, 62]. The authors in [2] analyzed 11 well-known biological networks and 9 well-known social networks for their $\mathfrak{C}_{\mathrm{Gromov}}$ values and found all but one network had a *statistically significant small* value of $\mathfrak{C}_{\mathrm{Gromov}}$. These references also describe implications of range of $\mathfrak{C}_{\mathrm{Gromov}}$ on the actual real-world applications of these networks. As mentioned in the following subsection, the Gromov-hyperbolicity measure is *fundamentally different* from the commonly used topological properties for a graph; for example, it is *neither* a hereditary *nor* a monotone property, is *not* the same as tree-width measure or other standard combinatorial properties that are commonly used in the computer science literature, and *not* necessarily a measure of closeness to tree topology.

4.1.2 Some clarifying remarks regarding Gromov-hyperbolicity measure

As pointed out in details by the authors in [26, Section 1.2.1], the Gromov-hyperbolicity measure $\mathfrak{C}_{\mathrm{Gromov}}$ enjoys many non-trivial topological characteristics. In particular, the authors in [26, Section 1.2.1] point out the following:

▷ $\mathfrak{C}_{\mathrm{Gromov}}$ is *not* a hereditary or monotone property since removal of nodes or edges may change the value of $\mathfrak{C}_{\mathrm{Gromov}}$ sharply.
▷ $\mathfrak{C}_{\mathrm{Gromov}}$ is *not* necessarily the same as tree-width measure (see also [2, 29]), or other standard combinatorial properties (*e.g.*, betweenness centrality, clustering coefficient, dense sub-graphs) that are commonly used in the computer science literature.
▷ "Close to hyperbolic topology" is *not* necessarily the same as "close to tree topology".

4.2 Geometric curvatures

In this section, we describe geometric curvatures of graphs by using correspondence with topological objects in higher dimension. The approach of using associations of sub-graphs with with topological objects in higher dimension has also been used in some previous papers such as [76] but our anomaly detection approach is quite different from them.

4.2.1 Basic topological concepts

We first review some basic concepts from topology; see introductory textbooks such as [36, 43] for further information. Although not absolutely necessary, the reader may find it useful to think of the underlying metric space as the r-dimensional real space $\mathbb{R}^r$ be for some integer $r > 1$.

▶ A subset $S \subseteq \mathbb{R}^r$ is *convex* if and only if for any $x, y \in S$, the *convex combination* of x and y is also in S.
▶ A set of $k+1$ points $x_0, \ldots, x_k \in \mathbb{R}^r$ are called *affinely independent* if and only if for all $\alpha_0, \ldots, \alpha_k \in \mathbb{R}$ $\sum_{j=0}^{k} \alpha_j x_j = 0$ and $\sum_{j=0}^{k} \alpha_j = 0$ implies $\alpha_0 = \cdots = \alpha_k = 0$.

▶ The *k-simplex* generated by a set of $k+1$ affinely independent points $x_0, \ldots, x_k \in \mathbb{R}^r$ is the subset $\mathcal{S}(x_0, \ldots, x_k)$ of $\mathbb{R}^r$ generated by *all* convex combinations of $x_0, \ldots, x_k$.

  ▷ Each $(\ell+1)$-subset $\{x_{i_0}, \ldots, x_{i_\ell}\} \subseteq \{x_0, \ldots, x_k\}$ defines the ℓ-simplex $\mathcal{S}(x_{i_0}, \ldots, x_{i_\ell})$ that is called a *face* of dimension ℓ (or a ℓ-face) of $\mathcal{S}(x_0, \ldots, x_k)$. A $(k-1)$-face, 1-face and 0-face is called a *facet*, an *edge* and a *node*, respectively.

▶ A (closed) *halfspace* is a set of points satisfying $\sum_{j=1}^r a_j x_j \le b$ for some $a_1, \ldots, a_r, b \in \mathbb{R}$. The convex set obtained by a bounded non-empty intersection of a finite number of halfspaces is called a *convex polytope* (*convex polygon* in two dimensions).

  ▷ If the intersection of a halfspace and a convex polytope is a subset of the halfspace then it is called a *face* of the polytope. Of particular interests are faces of dimensions $r-1$, 1 and 0, which are called *facets*, *edges* and *nodes* of the polytope, respectively.

▶ A *simplicial complex* (or just a complex) is a topological space constructed by the union of simplexes via topological associations.

4.2.2 Geometric curvature definitions

Informally, a *complex* is "glued" from nodes, edges and polygons via topological identification. We first define k-complex-based *Forman's combinatorial Ricci curvature* for elementary components (such as nodes, edges, triangles and higher-order cliques) as described in [12, 34, 75, 76], and then obtain a scalar curvature that takes an appropriate linear combination of these values (via Gauss-Bonnet type theorems, see for example [75, Sections 4.1–4.3] and the references therein) that correspond to the so-called *Euler characteristic* of the complex that is topologically associated with the given graph. In this paper, we consider such Euler characteristics of a graph to define geometric curvature.

To begin the topological association, we (topologically) associate a q-simplex with a $(q+1)$-clique $\mathcal{K}_{q+1}$; for example, 0-simplexes, 1-simplexes, 2-simplexes and 3-simplexes are associated with nodes, edges, 3-cycles (triangles) and 4-cliques, respectively. Next, we would also need the concept of an "order" of a simplex for more non-trivial topological association. Consider a p-face f^p of a q-simplex. An order d association of such a face, which we will denote by the notation f_d^p with the additional subscript d, is associated with a sub-graph of *at most* d nodes that is obtained by starting with $\mathcal{K}_{p+1}$ and then *optionally* replacing each edge by a path between the two nodes. For example,

– f_d^0 is a node of G for all $d \ge 1$.
– f_2^1 is an edge, and f_d^1 for $d > 2$ is a path having at most d nodes between two nodes adjacent in G.
– f_3^2 is a triangle (cycle of 3 nodes or a 3-cycle), and f_d^2 for $d > 3$ is obtained from 3 nodes by connecting every pair of nodes by a path such that the total number of nodes in the sub-graph is at most d.

Naturally, the higher the values of p and q are, the more complex are the topological associations. Let $\mathcal{F}_d^k$ be the set of all f_d^k's that are topologically associated. With such associations via p-faces of order d, the Euler characteristics of the graph $G = (V, E)$ and consequently the curvature can be defined as

$$\mathfrak{C}_d^p(G) \stackrel{\text{def}}{=} \sum_{k=0}^{p} (-1)^k \left| \mathcal{F}_d^k \right| \tag{1}$$

It is easy to see that both $\mathfrak{C}_d^0(G)$ and $\mathfrak{C}_d^1(G)$ are too simplistic to be of use in practice. Thus, we consider the next higher value of p in this paper, namely when $p = 2$. Letting $\mathcal{C}(G)$ denote the number of cycles of at most $d+1$ nodes in G, we get the measure

$$\mathfrak{C}_d^2(G) = |V| - |E| + |\mathcal{C}(G)|$$

Suitability of geometric curvature measures for real-world networks: The usefulness of geometric curvatures for real-world networks was demonstrated in publications such as [68, 75, 76].

5 Formalizations of two anomaly detection problems on networks

In this section, we formalize two versions of the anomaly detection problem on networks. An underlying assumption on the behind these formulations is that the graph adds/deletes *edges* only while keeping the same set of nodes.

5.1 Extremal anomaly detection for static networks

The problems in this subsection are motivated by a desire to quantify the extremal sensitivity of static networks. The basic decision question is: "*is there a subset among a set of prescribed edges whose deletion may change the network curvature significantly?*". This directly leads us to the following *decision* problem:

Problem name:	Extremal Anomaly Detection Problem
	$(\textsc{Eadp}_{\mathfrak{C}}(G, \widetilde{E}, \gamma))$
Input:	• A curvature measure $\mathfrak{C} : G \mapsto \mathbb{R}$
	• A connected graph $G = (V, E)$,
	an edge subset $\widetilde{E} \subseteq E$ such that $G \setminus \widetilde{E}$ is connected,
	and a real number $\gamma < \mathfrak{C}(G)$ (*resp.*, $\gamma > \mathfrak{C}(G)$)
Decision question:	is there an edge subset $\widehat{E} \subseteq \widetilde{E}$ such that $\mathfrak{C}(G \setminus \widehat{E}) \leq \gamma$
	(*resp.*, $\mathfrak{C}(G \setminus \widehat{E}) \geq \gamma$) ?
Optimization question:	if the answer to the decision question is "yes"
	then minimize $|\widehat{E}|$
Notation:	if the answer to the decision question is "yes" then
	the minimum possible value of $|\widehat{E}|$
	is denoted by $\mathsf{OPT}_{\textsc{Eadp}_{\mathfrak{C}}}(G, \widetilde{E}, \gamma)$

The following comments regarding the above formulation should be noted:

▷ For the case $\gamma < \mathfrak{C}(G)$ (*resp.*, $\gamma > \mathfrak{C}(G)$) we allow $\mathfrak{C}(G \setminus \widetilde{E}) > \gamma$ (*resp.*, $\mathfrak{C}(G \setminus \widetilde{E}) < \gamma$), thus $\widehat{E} = \widetilde{E}$ need *not* be a feasible solution at all.

▷ The curvature function is only defined for connected graphs, thus we require $G \setminus \widetilde{E}$ to be connected.

▷ The edges in $E \setminus \widetilde{E}$ can be thought of as "critical" edges needed for the functionality of the network. For example, in the context of inference of minimal biological networks from indirect experimental evidences [3, 4], the set of critical edges represent direct biochemical interactions with concrete evidence.

5.2 Targeted anomaly detection for dynamic networks

These problems are primarily motivated by change-point detections between two successive discrete time steps in dynamic networks [7, 51], but they can also be applied to static networks when a subset of the final desired network *is* known. Fig. 2 illustrates targeted anomaly detection for a dynamic biological network.

Problem name	:	Targeted Anomaly Detection Problem ($\mathrm{TADP}_{\mathfrak{C}}(G_1, G_2)$)
Input	:	• Two connected graphs $G_1 = (V, E_1)$ and $G_2 = (V, E_2)$ with $E_2 \subset E_1$ • A curvature measure $\mathfrak{C} : G \mapsto \mathbb{R}$
Valid solution	:	an edge subset $E_3 \subseteq E_1 \setminus E_2$ such that $\mathfrak{C}(G_1 \setminus E_3) = \mathfrak{C}(G_2)$.
Objective	:	*minimize* $|E_3|$.
Notation	:	the minimum value of $|E_3|$ is denoted by $\mathsf{OPT}_{\mathrm{TADP}_{\mathfrak{C}}}(G_1, G_2)$

6 Computational complexity of extremal anomaly detection problems

6.1 Geometric curvatures: computational complexity of $\mathrm{EADP}_{\mathfrak{C}_d^2}$

Theorem 1
(a) *The following statements hold for* $\mathrm{EADP}_{\mathfrak{C}_d^2}(G, \widetilde{E}, \gamma)$ *when* $\gamma > \mathfrak{C}_d^2(G)$:

(a1) *We can decide in polynomial time the answer to the decision question (i.e., if there exists any feasible solution $\widehat{E}$ or not).*

(a2) *If a feasible solution exists then the following results hold:*

(a2-1) *Computing $\mathsf{OPT}_{\mathrm{EADP}_{\mathfrak{C}_d^2}}(G, \widetilde{E}, \gamma)$ is NP-hard for all d that are multiple of 3.*

(a2-2) *If γ is sufficient larger than $\mathfrak{C}_d^2(G)$ then we can design an approximation algorithm that approximates both the cardinality of the minimal set of edges for deletion and the absolute difference between the two curvature values. More precisely, if $\gamma \geq \mathfrak{C}_d^2(G) + \left(\frac{1}{2} + \varepsilon\right)(2|\widetilde{E}| - |E|)$ for some $\varepsilon > 0$, then we can find in polynomial time a subset of edges $E_1 \subseteq \widetilde{E}$ such that*

$$|E_1| \leq 2\, \mathsf{OPT}_{\mathrm{EADP}_{\mathfrak{C}_d^2}}(G, \widetilde{E}, \gamma) \ \ \text{and} \ \ \frac{\mathfrak{C}_d^2(G \setminus E_1) - \mathfrak{C}_d^2(G)}{\gamma - \mathfrak{C}_d^2(G)} \geq \frac{4\varepsilon}{1 + 2\varepsilon}$$

(b) *The following statements hold for* $\mathrm{EADP}_{\mathfrak{C}_d^2}(G, \widetilde{E}, \gamma)$ *when* $\gamma < \mathfrak{C}_d^2(G)$:

(b1) *We can decide in polynomial time the answer to the decision question (i.e., if there exists any feasible solution $\widehat{E}$ or not).*

(b2) *If a feasible solution exists and γ is not too far below $\mathfrak{C}_d^2(G)$ then we can design an approximation algorithm that approximates both the cardinality of the minimal set of edges for deletion and the absolute difference between the two curvature values. More precisely, letting Δ denote the number of cycles of G of at most $d + 1$ nodes that contain at least one edge from $\widetilde{E}$,*

if $\gamma \geq \mathfrak{C}_d^2(G) - \frac{\Delta}{1+\varepsilon}$ for some $\varepsilon > 0$ then we can find in polynomial time a subset of edges $E_1 \subseteq \widetilde{E}$ such that

$$|E_1| \leq 2\,\mathsf{OPT}_{\mathrm{EADP}_{\mathfrak{C}_d^2}}(G, \widetilde{E}, \gamma) \quad \text{and} \quad \frac{\mathfrak{C}_d^2(G \setminus E_1) - \mathfrak{C}_d^2(G)}{\gamma - \mathfrak{C}_d^2(G)} \leq 1 - \varepsilon$$

(b*3*) *If $\gamma < \mathfrak{C}_d^2(G)$ then, even if $\gamma = \mathfrak{C}_d^2(G \setminus \widetilde{E})$ (i.e., a trivial feasible solution exists), computing $\mathsf{OPT}_{\mathrm{EADP}_{\mathfrak{C}_d^2}}(G, \widetilde{E}, \gamma)$ is at least as hard as computing $\mathrm{TADP}_{\mathfrak{C}_d^2}(G_1, G_2)$ and therefore **all** the hardness results for $\mathrm{TADP}_{\mathfrak{C}_d^2}(G_1, G_2)$ in Theorem 3 also apply to $\mathsf{OPT}_{\mathrm{EADP}_{\mathfrak{C}_d^2}}(G, \widetilde{E}, \gamma)$.*

6.1.1 Proof techniques and relevant comments regarding Theorem 1

(on proofs of (*a*1) and (*b*1)) After eliminating a few "easy-to-solve" sub-cases, we prove the remaining cases of (*a*1) and (*b*1) by reducing the feasibility questions to suitable minimum-cut problems; the reductions and proofs are somewhat different due to the nature of the objective function. It would of course be of interest if a single algorithm and proof can be found that covers both instances and, more importantly, if a direct and more efficient greedy algorithm can be found that *avoids* the maximum flow computation.

(on proofs of (*a*2-2) and (*b*2)) Our general approach to prove (*a*2-2) and (*b*2) is to formulate these problems as a series of (provably NP-hard and polynomially many) "constrained" minimum-cut problems. We start out with *two different* (but well-known) polytopes for the minimum cut problem (polytopes (4) and (4)′). Even though the polytope (4)′ is of exponential size for general graphs, it is of polynomial size for our particular minimum cut version and so we do *not* need to appeal to separation oracles for its efficient solution. We subsequently add extra constraints corresponding to a parameterized version of the minimization objective and solve the resulting augmented polytopes (polytopes (5) and (5)′) in polynomial time to get a fractional solution and use a simple deterministic rounding scheme to obtain the desired bounds.

▷ Our algorithmic approach uses a sequence of $\lceil \log_2(1 + |\widetilde{E}|) \rceil = O(\log |E|)$ linear-programming (LP) computations by using an obvious binary search over the relevant parameter range. It would be interesting to see if we can do the same using $O(1)$ LP computations.

▷ Is the factor 2 in "$|E_1| \leq 2\,\mathsf{OPT}_{\mathrm{EADP}_{\mathfrak{C}_d^2}}(G, \widetilde{E}, \gamma)$" an artifact of our specific rounding scheme around the threshold of $1/2$ and perhaps can be improved using a cleverer rounding scheme? This seems *unlikely* for the case when $\gamma < \mathfrak{C}_d^2(G)$ since the inapproximability results in (*b*3) include a $(2 - \varepsilon)$-inapproximability assuming the unique games conjecture is true. However, this possibility cannot be ruled out for the case when $\gamma > \mathfrak{C}_d^2(G)$ since we can only prove NP-hardness for this case.

▷ There are subtle but crucial differences between the rounding schemes for (**a2-2**) and (**b2**) that is essential to proving the desired bounds. To illustrate this, consider an edge e with a fractional value of $1/2$ for its corresponding variable. In the rounding scheme (6) of (**a2-2**) e will only *sometimes* be designated as a cut edge, whereas in the rounding scheme (6)′ of (**b2**) e will *always* be designated as a cut edge.

(on the bounds over γ in (a2-2)) If $|\widetilde{E}| \lessapprox \frac{1}{2}|E|$ then the condition on γ *is* redundant (*i.e.*, *always* holds). Thus indeed the 2-approximation is likely to hold *unconditionally* for *practical* applications of this problem since anomaly is supposed to be caused by a *large* change in curvature by a relatively *small* number of elementary components (edges in our cases).

Furthermore, if $|E| \leq 2|V|$ then the condition on γ always holds *irrespective* of the value of $|\widetilde{E}|$, and the *smaller* is $|\widetilde{E}|$ with respect to $|E|$ the *better* is our approximation of the curvature difference. As a general illustration, when $\varepsilon = 1/5$ the assumptions are $\gamma \geq \mathfrak{C}_d^2(G) + \frac{7}{10}(2|\widetilde{E}| - |E|)$, and the corresponding bounds are $|E_1| \leq 2\,\mathsf{OPT}_{\mathrm{EADP}_{\mathfrak{C}_d^2}}(G, \widetilde{E}, \gamma)$ and $\frac{\mathfrak{C}_d^2(G \setminus E_1) - \mathfrak{C}_d^2(G)}{\gamma - \mathfrak{C}_d^2(G)} \geq \frac{4}{7}$.

(on the hardness proof in (a2-1)) Our reduction is from the *densest-k-subgraph* (DkS$_3$) problem. We use the reduction from the CLIQUE problem to DkS$_3$ detailed by Feige and Seltser in [33] which shows that DkS$_3$ is NP-hard even if the degree of every node is at most 3. For convenience in doing calculations, we use the reduction of Feige and Seltser starting from the still NP-hard version of the CLIQUE problem where the input instances are $(n-4)$-regular n-node graphs. Pictorially, the reduction is illustrated in Fig. 3. Note that DkS$_3$ is *not* known to be $(1+\varepsilon)$-inapproximable assuming P$\neq$ NP (though it is likely to be), and thus our particular reduction cannot be generalized to $(1 + \varepsilon)$-inapproximability assuming P$\neq$ NP.

6.1.2 Proof of Theorem 1

Proof of (a1)

Let the notation $\mathcal{C}(H)$ denote the set of cycles having at most $d + 1$ nodes in a graph H. Assume $\Delta = |\mathcal{C}(G)|$ and let $\mathcal{C}(G) = \{\mathcal{F}_1, \mathcal{F}_2, \ldots, \mathcal{F}_\Delta\}$; thus $\mathfrak{C}_d^2(G) = n - m + \Delta$ where $|V| = n$ and $|E| = m$. Since d is fixed, $\Delta = O(n^d)$ and all the cycles in $\mathcal{C}(G)$ can be explicitly enumerated in polynomial ($O(n^d)$) time. Let $\mathcal{C}'(G) = \{\mathcal{F}_1, \mathcal{F}_2, \ldots, \mathcal{F}_{\Delta'}\} \subseteq \mathcal{C}(G)$ be the set of $\Delta' \leq \Delta$ cycles in $\mathcal{C}(G)$ that involve one of more edges from $\widetilde{E}$. An *overview* of the main steps in our proof for (a1) is as follows.

1. We identify sub-cases that are easy to solve.
2. For all remaining sub-cases, we reduce our problem to a standard (directed) minimum s-t cut problem such that the following statements hold:
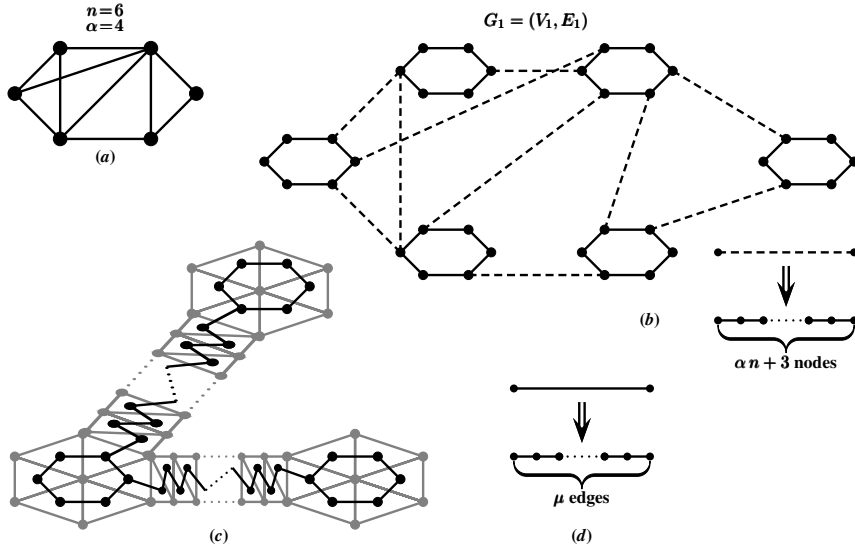
Fig. 3 Illustration of the reduction in the proof of Theorem 1(a2-1). (a) The original instance of the α-CLIQUE problem on an n-node graph. (b) Illustration of the NP-hardness reduction from α-CLIQUE to DkS$_3$ by Feige and Seltser [33]. (c) Illustration of associations of nodes with unique 3-cycles such that an edge between two nodes are adjacent correspond to sharing an unique edge of their associated 3-cycles. (d) Splitting of every edge of G_1 into a path of μ edges for the case when $d = 3\mu$ for some integer $\mu > 1$.

    ▷ The cut network can be constructed in polynomial time.

    ▷ There exists a feasible solution of $\mathrm{EADP}_{\mathfrak{C}_d^2}(G, \widetilde{E}, \gamma)$ if and

        only if the minimum cut value is at most $\Gamma \stackrel{\text{def}}{=} \mathfrak{C}_d^2(G) - \gamma + |\widetilde{E}|$.

Step 1. Identifying sub-cases that are easy to solve

    We first observe that the following sub-cases are easy to solve:

– If $\gamma > n - (m - |\widetilde{E}|) + \Delta$ then we can assert that there is no feasible solution. This is true because for any $E' \subseteq \widetilde{E}$ it is true that $\mathfrak{C}_d^2(G \setminus E')$ is at most $n - (m - |\widetilde{E}|) + \Delta$.

– If $\gamma \leq n - (m - |\widetilde{E}|) + \Delta$ and $\Delta' = 0$ then there exists a *trivial* optimal feasible solution of the following form:
        select any set of m_1 edges from $\widetilde{E}$ where m_1 is the least positive integer satisfying $n - (m_1 - |\widetilde{E}|) + \Delta \geq \gamma$.

Step 2. Solving all remaining sub-cases

    We assume that $\gamma \leq n - (m - |\widetilde{E}|) + \Delta$ and $\Delta' > 0$. Consider a subset $E_1 \subseteq \widetilde{E}$ of $m_1 = |E_1| \leq |\widetilde{E}|$ edges for deletion and suppose that removal of the edges in E_1 removes $\Delta_1 \leq \Delta'$ cycles from $\mathcal{C}'(G)$ (*i.e.*, $|\mathcal{C}'(G \setminus E_1)| = \Delta' - \Delta_1$). Then,

$$\mathfrak{C}_d^2(G \setminus E_1) = n - (m - m_1) + (\Delta - \Delta_1)$$
$$= n - m + \Delta + (m_1 - \Delta_1) = \mathfrak{C}_d^2(G) + (m_1 - \Delta_1) \quad (2)$$

and consequently one can observe that

$$\mathfrak{C}_d^2(G \setminus E_1) \geq \gamma \equiv m_1 - \Delta_1 \geq \gamma - \mathfrak{C}_d^2(G) \equiv \Delta_1 - m_1 \leq \mathfrak{C}_d^2(G) - \gamma$$
$$\equiv \Delta_1 + (m - m_1) \leq \mathfrak{C}_d^2(G) - \gamma + m \equiv \Delta_1 + (|\widetilde{E}| - m_1) \leq \mathfrak{C}_d^2(G) - \gamma + |\widetilde{E}| \stackrel{\text{def}}{=} \Gamma \tag{3}$$

Note that $\Gamma = \mathfrak{C}_d^2(G) - \gamma + |\widetilde{E}| = n - (m - |\widetilde{E}|) + \Delta - \gamma \geq 0$ and $|\widetilde{E}| - m_1$ is the number of edges in $\widetilde{E}$ that are *not* in E_1 and therefore not selected for deletion. Also, note that Γ is a quantity that depends on the problem instance only and *does not change if one or more edges are deleted*. Based on this interpretation, we construct the following instance (digraph) $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ of a (standard directed) minimum s-t cut problem (where $\mathrm{cap}(u, v)$ is the capacity of a directed edge (u, v)):

- The nodes in $\mathcal{V}$ are as follows: a source node s, a sink node t, a node (an "edge-node") u_e for every edge $e \in \widetilde{E}$ and a node (a "cycle-node") $u_{\mathcal{F}_i}$ for every cycle $\mathcal{F}_i \in \mathcal{C}'(G)$. The total number of nodes is therefore $O(|\widetilde{E}| + n^d)$, *i.e.*, polynomial in n.
- The directed edges in $\mathcal{E}$ and their corresponding capacities are as follows:
  - For every edge $e \in \widetilde{E}$, we have a directed edge (s, u_e) (an "edge-arc") of capacity $\mathrm{cap}(s, u_e) = 1$.
  - For every cycle $\mathcal{F}_i \in \mathcal{C}'(G)$, we have a directed edge (a "cycle-arc") $(u_{\mathcal{F}_i}, t)$ of capacity $\mathrm{cap}(u_{\mathcal{F}_i}, t) = 1$.
  - For every cycle $\mathcal{F}_i \in \mathcal{C}'(G)$ and every edge $e \in \widetilde{E}$ such that e is an edge of $\mathcal{F}_i$, we have a directed edge (an "ed-cy-arc", ed-cy-arc for short) $(u_e, u_{\mathcal{F}_i})$ of capacity $\mathrm{cap}(u_e, u_{\mathcal{F}_i}) = \infty$.

For an s-t cut $(\mathcal{S}, \mathcal{V} \setminus \mathcal{S})$ of $\mathcal{G}$ (where $s \in \mathcal{S}$ and $t \notin \mathcal{S}$), let $\mathrm{cut}(\mathcal{S}, \mathcal{V} \setminus \mathcal{S}) = \{(x, y) \,|\, x \in \mathcal{S}, y \notin \mathcal{S}\}$ and $\mathrm{cap}(\mathrm{cut}(\mathcal{S}, \mathcal{V} \setminus \mathcal{S})) = \sum_{(x,y) \in \mathrm{cut}(\mathcal{S}, \mathcal{V} \setminus \mathcal{S})} \mathrm{cap}(x, y)$ denote the edges in the cut and the capacity of the cut, respectively. It is well-known how to compute a *minimum s-t cut* of value $\Phi \stackrel{\text{def}}{=} \min_{\emptyset \subset \mathcal{S} \subset \mathcal{V}, \, s \in \mathcal{S}, \, t \notin \mathcal{S}} \{\mathrm{cap}(\mathrm{cut}(\mathcal{S}, \mathcal{V} \setminus \mathcal{S}))\}$ in polynomial time [23]. The following lemma proves part $(a1)$ of the theorem.

Lemma 1 *There exists any feasible solution of* $\mathrm{EADP}_{\mathfrak{C}_d^2}(G, \widetilde{E}, \gamma)$ *if and only if* $\Phi \leq \Gamma$. *Moreover, if* $(\mathcal{S}, \mathcal{V} \setminus \mathcal{S})$ *is a minimum s-t cut of* $\mathcal{G}$ *of value* $\Phi \leq \Gamma$ *then* $\widehat{E} = \{e \,|\, u_e \in \mathcal{S}\}$ *is a feasible solution for* $\mathrm{EADP}_{\mathfrak{C}_d^2}(G, \widetilde{E}, \gamma)$.

Proof. Suppose that there exists a feasible solution $E_1 \subseteq \widetilde{E}$ with $m_1 = |E_1|$ edges for $\mathrm{EADP}_{\mathfrak{C}_d^2}(G, \widetilde{E}, \gamma)$, and suppose that removal of the edges in E_1 removes Δ_1 cycles from $\mathcal{C}'(G)$. Consider the cut $(\mathcal{S}, \mathcal{V} \setminus \mathcal{S})$ where

$$\mathcal{S} = \{s\} \bigcup \{u_e \,|\, e \in E_1\} \bigcup \{u_{\mathcal{F}_i} \,|\, \mathcal{F}_i \text{ contains at least one edge from } E_1 \}$$

Note that *no* ed-cy-arc belongs to $\text{cut}(\mathcal{S}, \mathcal{V} \setminus \mathcal{S})$ and therefore

$$\text{cap}(\text{cut}(\mathcal{S}, \mathcal{V} \setminus \mathcal{S}))$$
$$= |\{(s, u_e) \mid e \notin E_1\}| + |\{(u_{\mathcal{F}_i}, t) \mid \mathcal{F}_i \text{ contains at least one edge from } E_1 \}|$$
$$= (|\widetilde{E}| - m_1) + \Delta_1$$

and thus by Inequality (3) we can conclude that

$$\mathfrak{C}_d^2(G \setminus E_1) \geq \gamma$$
$$\equiv \Delta_1 + (|\widetilde{E}| - m_1) \leq \Gamma \equiv \text{cap}(\text{cut}(\mathcal{S}, \mathcal{V}\setminus\mathcal{S})) \leq \Gamma \Rightarrow \Phi \leq \text{cap}(\text{cut}(\mathcal{S}, \mathcal{V}\setminus\mathcal{S})) \leq \Gamma$$

For the other direction, consider a minimum s-t cut $(\mathcal{S}, \mathcal{V} \setminus \mathcal{S})$ of $\mathcal{G}$ of value $\Phi \leq \Gamma$. Consider the solution $E_1 = \{e \mid u_e \in \mathcal{S}\} \subseteq \widetilde{E}$ for $\text{EADP}_{\mathfrak{C}_d^2}(G, \widetilde{E}, \gamma)$, and suppose that removal of the edges in E_1 removes Δ_1 cycles from $\mathcal{C}'(G)$. Since $\mathcal{G}$ admits a trivial s-t cut $(\{s\}, \mathcal{V} \setminus \{s\})$ of capacity $m_1 < \infty$, *no* ed-cy-arc can be an edge of *any* minimum s-t cut of $\mathcal{G}$, i.e., $\text{cut}(\mathcal{S}, \mathcal{V} \setminus \mathcal{S})$ contains only edge-arcs or cycle-arcs. Let $E_2 = \{\mathcal{F}_j \mid u_{\mathcal{F}_j} \in \mathcal{S}\}$. Consider an edge $e \in E_1$ and let $\mathcal{F}_j$ be a cycle in $\mathcal{C}'(G)$ containing e. Since $\text{cut}(\mathcal{S}, \mathcal{V} \setminus \mathcal{S})$ contains no ed-cy-arc, it does *not* contain the arc $(u_e, u_{\mathcal{F}_j})$. It thus follows that the cycle-node $u_{\mathcal{F}_j}$ must also belong to $\mathcal{S}$ and thus $|E_2| = \Delta_1$. Now note that

$$\Phi = |\{u_e \mid u_e \notin \mathcal{S}\}| + |\{u_{\mathcal{F}_j} \mid u_{\mathcal{F}_j} \in \mathcal{S}\}|$$
$$= (|\widetilde{E}| - |E_1|) + \Delta_1 \leq \Gamma = \mathfrak{C}_d^2(G) - \gamma + |\widetilde{E}|$$
$$\equiv \mathfrak{C}_d^2(G \setminus E_1) = \mathfrak{C}_d^2(G) + |E_1| - \Delta_1 \geq \gamma$$

❑

This completes a proof for $(a1)$.

Proof of $(a2\text{-}2)$ We will reuse the proof of $(a1)$ as appropriate. Let $\widehat{E} \subseteq \widetilde{E}$ be an optimal solution of the optimization version of $\text{EADP}_{\mathfrak{C}_d^2}(G, \widetilde{E}, \gamma)$ having $\text{OPT}_{\text{EADP}_{\mathfrak{C}_d^2}}(G, \widetilde{E}, \gamma)$ nodes. Note that $\text{OPT}_{\text{EADP}_{\mathfrak{C}_d^2}}(G, \widetilde{E}, \gamma) \in \{1, 2, \ldots, |\widetilde{E}|\}$ and thus in polynomial time we can "guess" every possible value of $\text{OPT}_{\text{EADP}_{\mathfrak{C}_d^2}}(G, \widetilde{E}, \gamma)$, solve the corresponding optimization problem with this additional constraint, and take the best of these solutions. In other words, it suffices if we can find, under the assumption that $\text{OPT}_{\text{EADP}_{\mathfrak{C}_d^2}}(G, \widetilde{E}, \gamma) = \kappa$ for some $\kappa \in \{1, 2, \ldots, |\widetilde{E}|\}$, find a solution $E_1 \subseteq \widetilde{E}$ satisfying the claims in $(a2\text{-}2)$. An *overview* of the main steps in our proof for $(a2\text{-}2)$ is as follows (where the comments are enclosed within a pair of $(*$ and $*))^3$.

    1. $(*$ *same as in* $(a1)$ $*)$

---

[3] For faster implementation, in the loop of Step 2 we can do binary search for the least possible κ over the range $\{1, 2, \ldots, |\widetilde{E}|\}$ for which the polytope's optimal solution value is at most Γ, requiring $\lceil \log_2(1 + |\widetilde{E}|) \rceil$ iterations instead of $|\widetilde{E}|$ iterations. For clarity, we omit such obvious improvements.

We identify sub-cases whose optimal solutions are easy to find.

Following steps apply only to all remaining sub-cases.

2. **for** $\kappa = 1, 2, \ldots, |\widetilde{E}|$ **do**   (* assume $\mathsf{OPT}_{\mathrm{EADP}_{\mathfrak{c}_d^2}}(G, \widetilde{E}, \gamma) = \kappa$ *)

2a.   (* *as in* (*a*1) *but with an additional constraint* *)

we reduce our problem to a (directed) minimum s-t cut problem

with the following additional constraint

▶ the number of edges to be deleted from $\widetilde{E}$ is κ

such that the following statements hold:

▷ The cut network can be constructed in polynomial time.

▷ There exists a feasible solution of $\mathrm{EADP}_{\mathfrak{c}_d^2}(G, \widetilde{E}, \gamma)$

if and only if the minimum cut value is

at most $\Gamma \overset{\mathrm{def}}{=} \mathfrak{C}_d^2(G) - \gamma + |\widetilde{E}|$.

2b.   find an extreme-point optimal solution for an appropriate

polytope for the constrained minimum cut problem

in polynomial time.

2c.   **if** the optimal objective value is at most Γ **then**

2c(i).   carefully convert relevant fractional values in the solution

to integral values to get a solution in polynomial time.

3. Return the best among all solutions found in Step **2**

as the desired solution.

Step 2*b*. Formulating an appropriate polytope for the constrained minimum cut problem

We showed in the proof of part (*a*1) that the feasibility problem can be reduced to finding a minimum s-t cut of the directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. Notice that $\mathcal{G}$ is acyclic, and every path between s and t has *exactly three* directed edges, namely an edge-arc followed by a ed-cy-arc followed by a cycle-arc. The minimum s-t cut problem for a graph has a well-known associated convex polytope of *polynomial* size (*e.g.*, see [72, pp. 98-99]). Letting p_β to be the variable corresponding to each node $\beta \in \mathcal{V}$, and d_α to be the variable associated with the edge $\alpha \in \mathcal{E}$, this minimum s-t cut polytope for the graph $\mathcal{G}$ is as follows:

minimize $\sum_{\alpha \in \mathcal{E}} \mathrm{cap}(\alpha) d_\alpha$

$= \sum_{\alpha \in \mathcal{E},\ \alpha \text{ is not ed-cy-arc}} d_\alpha + \sum_{\alpha \in \mathcal{E},\ \alpha \text{ is ed-cy-arc}} \infty \times d_\alpha$

subject to $d_\alpha \geq p_\beta - p_\xi$ for every edge $\alpha = (\beta, \xi) \in \mathcal{E}$

$p_s - p_t \geq 1$

$0 \leq p_\beta \leq 1$ for every node $\beta \in \mathcal{V}$

$0 \leq d_\alpha \leq 1$ for every edge $\alpha \in \mathcal{E}$

(4)

It is well-known that all extreme-point solutions of (4) are integral. An integral solution of (4) generates a s-t cut $(\mathcal{S}, \mathcal{V} \setminus \mathcal{S})$ by letting $\mathcal{S} = \{\beta \mid p_\beta = 1\}$ and $\mathrm{cut}(\mathcal{S}, \mathcal{V} \setminus \mathcal{S}) = \{\alpha \mid d_\alpha = 1\}$. For our case, we have an additional constraint in that the number of edges to be deleted from $\widetilde{E}$ is κ, which motivates us to formulate the following polytope for our problem:

$$
\begin{aligned}
\text{minimize} \quad & \sum_{\alpha \in \mathcal{E}} \mathrm{cap}(\alpha) d_\alpha \\
& = \sum_{\alpha \in \mathcal{E}, \ \alpha \text{ is not ed-cy-arc}} d_\alpha + \sum_{\alpha \in \mathcal{E}, \ \alpha \text{ is ed-cy-arc}} \infty \times d_\alpha \\
\text{subject to} \quad & d_\alpha \geq p_\beta - p_\xi \qquad \text{for every edge } \alpha = (\beta, \xi) \in \mathcal{E} \\
& p_s - p_t \geq 1 \\
& 0 \leq p_\beta \leq 1 \qquad\quad \text{for every node } \beta \in \mathcal{V} \\
& 0 \leq d_\alpha \leq 1 \qquad\quad \text{for every edge } \alpha \in \mathcal{E} \\
& \sum_{u_e \in \mathcal{V}} p_{u_e} = \kappa
\end{aligned}
$$

$$\tag{5}$$

Let $\mathsf{OPT}_{(5)}$ denote the optimal objective value of (5).

Lemma 2 $\mathsf{OPT}_{(5)} \leq \Gamma$.

Proof. Suppose that removal of the edges in the optimal solution $\widehat{E}$ removes $\widehat{\Delta} \leq \Delta'$ cycles from $\mathcal{C}'(G)$. We construct the following solution of (5) with respect to the optimal solution $\widehat{E}$ of $\mathrm{EADP}_{\mathfrak{C}_d^2}(G, \widetilde{E}, \gamma)$ having $|\widehat{E}| = \kappa$ nodes:

$$
\mathcal{S} = \{s\} \bigcup \{u_e \mid e \in \widehat{E}\} \bigcup \left\{ u_{\mathcal{F}_i} \mid \mathcal{F}_i \text{ contains at least one edge from } \widehat{E} \right\}
$$

$$
p_\beta = \begin{cases} 1, & \text{if } \beta \in \mathcal{S} \\ 0, & \text{otherwise} \end{cases} \qquad d_\alpha = \begin{cases} 1, & \text{if } \alpha \in \mathrm{cut}(\mathcal{S}, \mathcal{V} \setminus \mathcal{S}) \\ 0, & \text{otherwise} \end{cases}
$$

It can be verified as follows that this is indeed a feasible solution of (5):

- Since $p_s = 1$ and $p_t = 0$, it follows that $p_s - p_t \geq 1$ is satisfied.
- No ed-cy-arc belongs to $\mathrm{cut}(\mathcal{S}, \mathcal{V} \setminus \mathcal{S})$. Thus, if $\alpha = (\beta, \xi)$ is an ed-cy-arc then $d_\alpha = 0$ and it is not the case that $p_\beta = 1$ and $p_\xi = 0$. Thus for every ed-cy-arc α the constraint $d_\alpha \geq p_\beta - p_\xi$ is satisfied.
- Consider an edge-arc $\alpha = (s, u_e)$; note that $p_s = 1$. If $u_e \in \mathcal{S}$ that $p_{u_e} = 1$ and $d_\alpha = 0$, otherwise $p_{u_e} = 0$ and $d_\alpha = 1$. In both cases, the constraint $d_\alpha \geq p_\beta - p_\xi$ is satisfied. The case of a cycle-arc is similar.
- The constraint $\sum_{u_e \in \mathcal{V}} p_{u_e} = \kappa$ is *trivially* satisfied since $|\widetilde{E}| = \kappa$ by our assumption.

Note that $\mathrm{cut}(\mathcal{S}, \mathcal{V} \setminus \mathcal{S})$ does *not* contain any ed-cy-arcs. Thus, the objective value of this solution is

$$
\sum_{a \in \mathcal{E}} d_a = \sum_{a \in \mathrm{cut}(\mathcal{S}, \mathcal{V} \setminus \mathcal{S})} d_a = |\{u_e \mid u_e \notin \mathcal{S}\}| + |\{u_{\mathcal{F}_j} \mid u_{\mathcal{F}_j} \in \mathcal{S}\}| = (|\widetilde{E}| - |\widehat{E}|) + \widehat{\Delta} \leq \Gamma
$$

where the last inequality follows by (3) since $\mathfrak{C}_d^2(G \setminus \widehat{E}) \geq \gamma$. $\qquad\square$

Step 2c. Post-processing fractional values in the polytope solution

Given a polynomial-time obtainable optimal solution values $\left\{ d_\alpha^*, p_\beta^* \,|\, \alpha \in \mathcal{E}, \beta \in \mathcal{V} \right\}$ of the variables in (5), consider the following simple rounding procedure, the corresponding cut $(\mathcal{S}, \mathcal{V} \backslash \mathcal{S})$ of $\mathcal{G}$, and the corresponding solution $E_1 \subseteq \widetilde{E}$ of $\mathrm{EADP}_{\mathfrak{C}_d^2}(G, \widetilde{E}, \gamma)$:

$$\hat{p}_\beta = \begin{cases} 1, \text{ if } p_\beta^* \geq {}^1\!/\!_2 \\ 0, \text{ otherwise} \end{cases} \quad \mathcal{S} = \{ \beta \in \mathcal{V} \,|\, \hat{p}_\beta = 1 \} \quad E_1 = \{ e \,|\, u_e \in \mathcal{S} \} \quad (6)$$

Note that in inequalities $p_s - p_t \geq 1$, $0 \leq p_s \leq 1$ and $0 \leq p_t \leq 1$ ensures that $p_s^* = 1$ and $p_t^* = 0$.

Lemma 3 $|E_1| \leq 2\,\kappa$.

Proof. $|E_1| = |\{ u_e \,|\, p_{u_e}^* \geq {}^1\!/\!_2 \}| \leq 2 \sum_{u_e \in \mathcal{V}} p_{u_e}^* = 2\,\kappa$. $\qquad\qquad$ ❏

Lemma 4 $\mathrm{cap}(\mathrm{cut}(\mathcal{S}, \mathcal{V} \setminus \mathcal{S})) \leq 2\,\mathsf{OPT}_{(5)} \leq 2\,\Gamma$.

Proof. Since $\mathrm{cap}(\alpha) = \infty$ and $\mathsf{OPT}_{(5)} \leq \Gamma < \infty$, $d_\alpha^* = 0$ for any ed-cy-arc $\alpha = (u_e, u_{\mathcal{F}_j})$, and thus $p_{u_e}^* \leq p_{u_{\mathcal{F}_j}}^*$ for such an edge. It therefore follows that

$$p_{u_e}^* \geq {}^1\!/\!_2 \Rightarrow p_{u_{\mathcal{F}_j}}^* \geq {}^1\!/\!_2 \quad \equiv \quad \hat{p}_{u_e} = 1 \Rightarrow \hat{p}_{u_{\mathcal{F}_j}} = 1$$

Thus, no ed-cy-arc belongs to $\mathrm{cut}(\mathcal{S}, \mathcal{V} \setminus \mathcal{S})$. Thus using Lemma 2 it follows that

$$\begin{aligned}
\mathrm{cap}(\mathrm{cut}(\mathcal{S}, \mathcal{V} \setminus \mathcal{S})) &= |\{ (s, u_e) \,|\, \hat{p}_{u_e} = 0 \}| + |\{ (u_{\mathcal{F}_j}, t) \,|\, \hat{p}_{u_{\mathcal{F}_j}} = 1 \}| \\
&= |\{ (s, u_e) \,|\, p_{u_e}^* < {}^1\!/\!_2 \}| + |\{ (u_{\mathcal{F}_j}, t) \,|\, p_{u_{\mathcal{F}_j}}^* \geq {}^1\!/\!_2 \}| \\
&\leq 2 \sum_{p_{u_e}^* < {}^1\!/\!_2} (p_s^* - p_{u_e}^*) + 2 \sum_{p_{\mathcal{F}_j}^* \geq {}^1\!/\!_2} (p_{\mathcal{F}_j}^* - p_t^*) \\
&\leq 2 \sum_{p_{u_e}^* < {}^1\!/\!_2} d_{s, p_{u_e}}^* + 2 \sum_{p_{\mathcal{F}_j}^* \geq {}^1\!/\!_2} d_{s, p_{\mathcal{F}_j}}^* < 2 \sum_{\alpha \in \mathcal{E}} \mathrm{cap}(\alpha) d_\alpha^* \leq 2\,\Gamma
\end{aligned}$$

$\qquad\qquad$ ❏

Since no ed-cy-arc belongs to $\mathrm{cut}(\mathcal{S}, \mathcal{V} \setminus \mathcal{S})$, if an edge $e \in E_1$ is involved in a cycle $\mathcal{F}_j \in \mathcal{C}'(G)$ then it must be the case that $(u_e, u_{\mathcal{F}_j}) \notin \mathrm{cut}(\mathcal{S}, \mathcal{V} \setminus \mathcal{S})$. Thus, letting $m_1 = |E_1|$ and $\Delta_1 = |\{ \mathcal{F}_j \in \mathcal{C}'(G) \,|\, u_{\mathcal{F}_j} \in \mathcal{S} \}|$, the claimed bound on $\mathfrak{C}_d^2(G \setminus E_1)$ can be shown as follows using Lemma 4:

$$\begin{aligned}
\mathrm{cap}(\mathrm{cut}(\mathcal{S}, \mathcal{V} \setminus \mathcal{S})) &= (m - m_1) + \Delta_1 \leq 2\,\Gamma = 2\,\mathfrak{C}_d^2(G) - 2\,\gamma + 2\,|\widetilde{E}| \\
\Rightarrow \mathfrak{C}_d^2(G \setminus E_1) &= \mathfrak{C}_d^2(G) + m_1 - \Delta_1 \geq 2\,\gamma - \mathfrak{C}_d^2(G) - (2\,|\widetilde{E}| - m), \quad \text{by (2)} \\
\Rightarrow \frac{\mathfrak{C}_d^2(G \setminus E_1) - \mathfrak{C}_d^2(G)}{\gamma - \mathfrak{C}_d^2(G)} &\geq 2 - \frac{2\,|\widetilde{E}| - m}{\gamma - \mathfrak{C}_d^2(G)} \geq 2 - \frac{1}{\frac{1}{2} + \varepsilon} = \frac{4\varepsilon}{1 + 2\varepsilon}
\end{aligned}$$

(a2-1) The decision version of computing $\mathsf{OPT}_{\mathrm{EADP}_{\mathfrak{c}_d^2}}(G, \widetilde{E}, \gamma)$ is as follows: "given an instance $\mathrm{EADP}_{\mathfrak{c}_d^2}(G, \widetilde{E}, \gamma)$ and an integer $\kappa > 0$, is there a solution $\widehat{E} \subseteq \widetilde{E}$ satisfying $|\widehat{E}| \leq \kappa$?". We first consider the case of $d = 3$. We will reduce from the decision version of the $\mathrm{D}k\mathrm{S}_3$ problem which is defined as follows.

Definition 3 ($\mathrm{D}k\mathrm{S}_3$ problem) Given an undirected graph $G_1 = (V_1, E_1)$ where the degree of every node is either 2 or 3 and two integers k and t, is there a (node-induced) subgraph of G_1 that has k nodes and at least t edges?

Assuming that their reduction is done from the clique problem on a $(n-4)$-regular n-node graph (which is NP-hard [20]), the proof of Feige and Seltser in [33] shows that $\mathrm{D}k\mathrm{S}_3$ is NP-complete for the following parameter values (for some integer $\sqrt{n} < \alpha \leq n - 4$):

$$|V_1| = n^2 + (\alpha\, n + 1)\left(\frac{n\,(n-4)}{2}\right), \quad |E_1| = |V_1| + \frac{n\,(n-4)}{2}$$

$$k = \alpha\, n + \binom{\alpha}{2}(\alpha\, n + 1), \quad t = \alpha\, n + \binom{\alpha}{2}(\alpha\, n + 2)$$

We briefly review the reduction of Feige and Seltser in [33] as needed from our purpose. Their reduction is from the α-CLIQUE problem which is defined as follows.

Definition 4 (α-CLIQUE problem) Given a graph of n nodes, does there exist a clique (complete subgraph) of size α?

Given an instance of α-CLIQUE, they create an instance $G_1 = (V_1, E_1)$ of $\mathrm{D}k\mathrm{S}_3$ (with the parameter values shown above) in which every node is replaced by a cycle of n edges and an edge between two nodes is replaces by a path of length $\alpha\, n + 3$ between two unique nodes of the two cycles corresponding to the two nodes (see Fig. 3 (a)–(b) for an illustration). Given such an instance of $\mathrm{D}k\mathrm{S}_3$ with $V_1 = \{u_1, \ldots, u_{|V_1|}\}$ and $E_1 = \{a_1, \ldots, a_{|E_1|}\}$, we create an instance of $\mathrm{EADP}_{\mathfrak{c}_3^2}(G, \widetilde{E}, \gamma)$ as follows:

- We associate each node $u_i \in V_1$ with a triangle (the "node triangle") $\mathcal{L}_i$ of 3 nodes in V such that every edge $\{u_i, u_j\} \in E_1$ is mapped to a *unique* edge (the "shared edge") $e_{u_i, u_j} \in E$ that is shared by $\mathcal{L}_i$ and $\mathcal{L}_j$ (see Fig. 3 (c)). Since in the reduction of Feige and Seltser [33] all nodes have degree 2 or 3 and two degree 3 nodes do not share more than one edge such a node-triangle association is possible. We set $\widetilde{E}$ to be the set of *all* shared edges; note that $|\widetilde{E}| = |E_1|$. Let $\mathcal{L} = \{v_1, v_2, \ldots\ldots\}$ be the set of all nodes in the that appear in any node triangle; note that $|\mathcal{L}| < 3\,|V_1|$.
- To maintain connectivity after all edges in $\widetilde{E}$ are deleted, we introduce $3|\mathcal{L}| + 1$ new nodes $\{w_0\} \cup \{w_{i,j} \,|\, i \in \{1, 2, \ldots, |\mathcal{L}|\}, j \in \{1, 2, 3\}\}$ and $4\,|\mathcal{L}|$ new edges

$$\Big\{ \{w_0, w_{j,1}\}, \{w_{j,1}, w_{j,2}\}, \{w_{j,2}, w_{j,3}\}, \{w_{j,3}, v_j\} \,|\, j \in \{1, 2, \ldots, |\mathcal{L}|\} \Big\}$$

– We set $\gamma = \mathfrak{C}_3^2(G) + (t - k) = \mathfrak{C}_3^2(G) + \binom{\alpha}{2}$.

First, we show that $\mathrm{EADP}_{\mathfrak{C}_3^2}(G, \widetilde{E}, \gamma)$ indeed *has* a trivial feasible solution, namely a solution that contains all the edges from $\widetilde{E}$. The number of triangles Δ' that include one or more edges from $\widetilde{E}$ is precisely $|V_1|$ and thus using (2) we get:

$$\mathfrak{C}_3^2(G \setminus \widetilde{E}) = \mathfrak{C}_3^2(G) + |\widetilde{E}| - |\Delta'| = \mathfrak{C}_3^2(G) + |E_1| - |V_1|$$
$$= \mathfrak{C}_3^2(G) + \frac{n(n-4)}{2} > \mathfrak{C}_3^2(G) + \binom{\alpha}{2} = \gamma$$

where the last inequality follows since $\alpha \leq n - 4$. The following lemma completes our proof.

Lemma 5 *G_1 has a subgraph of k nodes and at least t edges if and only if the instance of $\mathrm{EADP}_{\mathfrak{C}_3^2}(G, \widetilde{E}, \gamma)$ constructed above has a solution $\widehat{E} \subseteq \widetilde{E}$ satisfying $|\widehat{E}| \leq t$.*

Proof. Suppose that G_1 has k nodes $u_1, u_2, \ldots, u_k$ such that the subgraph H_1 induced by these nodes has $t' \geq t$ edges. Remove an arbitrary set of $t' - t$ edges from H_1 to obtain a subgraph $H_1' = (V_1', E_1')$, and let $\widehat{E} = \{e_{u_i,u_j} \mid i, j \in \{1, 2, \ldots, k\}, \{u_i, u_j\} \in E_1'\}$. Obviously, $|\widehat{E}| = t$. Consider the triangle $\mathcal{L}_i$ corresponding to a node $u_i \in \{u_1, u_k, \ldots, u_k\}$, and let $I(\mathcal{L}_i)$ be the 0-1 *indicator variable* denoting if $\mathcal{L}_i$ is eliminated by removing the edges in $\widehat{E}$, i.e., $I(\mathcal{L}_i) = 1$ (resp., $I(\mathcal{L}_i) = 0$) if and only if $\mathcal{L}_i$ *is* eliminated (resp., is *not* eliminated) by removing the edges in $\widehat{E}$. Note that the triangle $\mathcal{L}_i$ gets removed if and only if there exists another node $u_j \in \{u_1, u_k, \ldots, u_k\}$ such that $\{u_i, u_j\} \in E_1$. Thus, the total number of triangles eliminated by removing the edges in $\widehat{E}$ is at most $\sum_{i=1}^{k} I(\mathcal{L}_i) \leq k$ and consequently

$$\mathfrak{C}_3^2(G \setminus E') = \mathfrak{C}_3^2(G) + |\widehat{E}| - \sum_{i=1}^{k} I(\mathcal{L}_i) \geq \mathfrak{C}_3^2(G) + t - k = \gamma$$

Conversely, suppose that the instance of $\mathrm{EADP}_{\mathfrak{C}_3^2}(G, \widetilde{E}, \gamma)$ has a solution $\widehat{E} \subseteq \widetilde{E}$ satisfying $|\widehat{E}| = \widehat{t} \leq t$. Let
$V_1' = \{u_j \mid \mathcal{L}_j \text{ is removed by removing one of more edges from } \widehat{E}\}$. Using (2) we get

$$\mathfrak{C}_3^2(G \setminus \widehat{E}) \geq \gamma = \mathfrak{C}_3^2(G) + t - k \Rightarrow \widehat{t} - |V_1'| \geq t - k \qquad (7)$$

Let $H_1' = (V_1', E_1')$ be the subgraph of G_1 induced by the nodes in V_1'. Clearly, $|E_1'| \geq \widehat{t}$. If $|V_1| < k$ then we use the following procedure to add $k - |V_1'|$ nodes:

$\quad V_1'' \leftarrow V_1'$
$\quad$**while** $|V_1''| \neq k$ **do**
$\quad\quad$select a node $u_j \notin V_1''$ connected to one or more nodes in V_1'',
$\quad\quad\quad$and add u_j to V_1''

Let $H_1'' = (V_1'', E_1'')$ be the subgraph of G_1 induced by the nodes in V_1''. Note that $|V_1''| = k$ and $|E_1''| \geq |E_1'| + (k - |V_1'|)$, and thus using (7) we get

$$|E_1''| \geq |E_1'| + (k - |V_1'|) \geq \widehat{t} + (k - |V_1'|) \geq t$$

<div align="right">❑</div>

This concludes the proof for $d = 3$. For the case when $d = 3\mu$ for some integer $\mu > 1$, the same reduction can be used provide we split *every* edge of G_1 into a path of length μ by using new $\mu - 1$ nodes (see Fig. 3 (d)).

(b1) and (b2) We *will* reuse the notations used in the proof of (a). We modify the proof and the proof technique in **(a1)** for the proof of **(b1)**. We now observe that the following sub-cases are easy to solve:

- If $\gamma < n - m + 1 + \Delta - \Delta'$ then we can assert that there is no feasible solution. This is true because for any $E' \subseteq \widetilde{E}$ it is true that $\mathfrak{C}_d^2(G \setminus E')$ is at least $n - (m - 1) + (\Delta - \Delta')$.
- If $\gamma \geq n - m + 1 + \Delta - \Delta'$ and $\Delta' = 0$ then there exists a *trivial* optimal feasible solution of the following form: *select any set of m_1 edges from $\widetilde{E}$ where m_1 is the largest positive integer satisfying $n - m_1 + 1 + \Delta \leq \gamma$.*

Thus, we assume that $\gamma \geq n - m + 1 + \Delta - \Delta'$ and $\Delta' > 0$. (2) still holds, but (3) is now rewritten as (note that $\Gamma > 0$):

$$\mathfrak{C}_d^2(G \setminus E_1) \leq \gamma \equiv m_1 - \Delta_1 \leq \gamma - \mathfrak{C}_d^2(G)$$

$$\equiv m_1 + (\Delta' - \Delta_1) \leq \gamma - \mathfrak{C}_d^2(G) + \Delta' \stackrel{\text{def}}{=} \Gamma \quad (3)'$$

The nodes in the di-graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ are same as before, but the directed edges are modified as follows:

- For every edge $e \in \widetilde{E}$, we have an edge (u_e, t) (an "edge-arc") of capacity $\text{cap}(u_e, t) = 1$.
- For every cycle $\mathcal{F}_i \in \mathcal{C}'(G)$, we have an edge (a "cycle-arc") $(s, u_{\mathcal{F}_i})$ of capacity $\text{cap}(s, u_{\mathcal{F}_i}) = 1$.
- For every cycle $\mathcal{F}_i \in \mathcal{C}'(G)$ and every edge $e \in \widetilde{E}$ such that e is an edge of $\mathcal{F}_i$, we have a directed edge (an "cycle-edge-arc", cy-ed-arc for short) $(u_{\mathcal{F}_i}, u_e)$ of capacity $\text{cap}(u_{\mathcal{F}_i}, u_e) = \infty$.

Corresponding to a feasible solution E_1 of m_1 edges for $\text{EADP}_{\mathfrak{C}_d^2}(G, \widetilde{E}, \gamma)$ that removes Δ_1 cycles, exactly the same cut $(\mathcal{S}, \mathcal{V} \setminus \mathcal{S})$ described before includes *no* cy-ed-arcs and has a capacity of

$$\text{cap}(\text{cut}(\mathcal{S}, \mathcal{V} \setminus \mathcal{S})) = |\{(s, u_{\mathcal{F}_i}) \,|\, \mathcal{F}_i \text{ does not contain one or mores edges from } E_1 \}|$$
$$+ |\{(u_e, t) \,|\, e \in E_1\}|$$
$$= (\Delta' - \Delta_1) + m_1$$

Therefore $\mathfrak{C}_d^2(G \setminus E_1) \leq \gamma$ implies $\text{cap}(\text{cut}(\mathcal{S}, \mathcal{V} \setminus \mathcal{S})) \leq \Gamma$, as desired. Conversely, given a minimum s-t cut $(\mathcal{S}, \mathcal{V} \setminus \mathcal{S})$ of $\mathcal{G}$ of value $\Phi \leq \Gamma$, we consider

the solution $E_1 = \{e \mid u_e \in \mathcal{S}\}$ for $\text{EADP}_{\mathfrak{C}_{\tilde{d}}^2}(G, \tilde{E}, \gamma)$. Let $\Psi = \{\mathcal{F}_j \mid u_{\mathcal{F}_j} \in \mathcal{S}\}$ and let Υ be the cycles from $\mathcal{C}'(G)$ that are removed by deletion of the edges in E_1. Since *no* cy-ed-arc (of infinite capacity) can be an edge of the minimum s-t cut $(\mathcal{S}, \mathcal{V} \setminus \mathcal{S})$, Ψ is a subset of Υ. We therefore have

$$\Phi = |\{u_{\mathcal{F}_j} \mid u_{\mathcal{F}_j} \notin \mathcal{S}\}| + |\{u_e \mid u_e \in \mathcal{S}\}|$$
$$= (\Delta' - |\Psi|) + |E_1| \leq \Gamma \;\Rightarrow\; (\Delta' - |\Upsilon|) + |E_1| \leq \Gamma$$

and the last inequality implies $\mathfrak{C}_d^2(G \setminus E_1) \leq \gamma$.

This completes a proof for $(b1)$. We now prove $(b2)$. We use an approach similar to that in $(a2)$ but with a *different polytope* for the minimum s-t cut of $\mathcal{G}$. Let $\mathcal{P}$ be the set of all possible s-t paths in $\mathcal{G}$. Then, an alternate polytope for the minimum s-t cut is as follows (cf. see (20.2) in [72, p. 168]):

$$\begin{array}{lll} \text{minimize} & \sum_{\alpha \in \mathcal{E}} \text{cap}(\alpha) d_\alpha & \\ \text{subject to} & \sum_{\alpha \in p} d_\alpha \geq 1 & \text{for every } s\text{-}t \text{ path } p \in \mathcal{P} \\ & 0 \leq d_\alpha \leq 1 & \text{for every edge } \alpha \in \mathcal{E} \end{array} \qquad (4)'$$

An integral solution of $(4)'$ generates a s-t cut $(\mathcal{S}, \mathcal{V} \setminus \mathcal{S})$ by letting $\text{cut}(\mathcal{S}, \mathcal{V} \setminus \mathcal{S}) = \{\alpha \mid d_\alpha = 1\}$. Since the capacity of any cy-ed-arc in ∞, $\text{cut}(\mathcal{S}, \mathcal{V} \setminus \mathcal{S})$ contains only cycle-arcs or edge-arcs, and the number of edge-arcs in $\text{cut}(\mathcal{S}, \mathcal{V} \setminus \mathcal{S})$ for an integral solution is precisely the number of edge-nodes in $\mathcal{S}$. This motivates us to formulate the following polytope for our problem to ensure that integral solutions constrain the number of edges to be deleted from $\tilde{E}$ to be κ:

$$\begin{array}{lll} \text{minimize} & \sum_{\alpha \in \mathcal{E}} \text{cap}(\alpha) d_\alpha & \\ \text{subject to} & \sum_{\alpha \in p} d_\alpha \geq 1 & \text{for every } s\text{-}t \text{ path } p \in \mathcal{P} \\ & 0 \leq d_\alpha \leq 1 & \text{for every edge } \alpha \in \mathcal{E} \\ & \sum_{e \in \tilde{E}} d_{(u_e, t)} = \kappa & \end{array} \qquad (5)'$$

For our problem, $|\mathcal{P}| < \binom{|\mathcal{V}|}{3}$ and thus $(5)'$ can be solved in polynomial time. Let $\text{OPT}_{(5)'}$ denote the optimal objective value of $(5)'$. It is very easy to see that $\text{OPT}_{(5)'} \leq \Gamma$: assuming that deletion of the κ edges in the optimal solution $\hat{E}$ removes $\hat{\Delta}$ cycles from $\mathcal{C}'(G)$, we set $d_\alpha = \begin{cases} 1, \alpha = (u_{\mathcal{F}_j}, u_e), e \in \hat{E} \\ 0, \text{otherwise} \end{cases}$
to construct a feasible solution of $(5)'$ of objective value

$$\sum_{\alpha \in \mathcal{E}} d_\alpha = |\{u_{\mathcal{F}_j} \mid d_{(s, u_{\mathcal{F}_j})} = 1\}| + |\{u_e \mid d_{(u_e, t)} = 1\}| = (\Delta' - \hat{\Delta}) + |\tilde{E}| \leq \Gamma$$

where the last inequality follows by $(3)'$ since $\mathfrak{C}_d^2(G \setminus \hat{E}) \leq \gamma$. Note that the constraint $\sum_{e \in \tilde{E}} d_{(u_e, t)} = \kappa$ is satisfied since $\sum_{e \in \tilde{E}} d_{(u_e, t)} = |\{d_{(u_e, t)} \mid d_{(u_e, t)} = 1\}| = |\{e \mid e \in \hat{E}\}| = \kappa$.

Given a polynomial-time obtainable optimal solution values $\{d_\alpha^* \mid \alpha \in \mathcal{E}\}$ of the variables in $(5)'$, consider the following simple rounding procedure, the

corresponding cut $(\mathcal{S}, \mathcal{V} \setminus \mathcal{S})$ of $\mathcal{G}$, and the corresponding solution $E_1 \subseteq \widetilde{E}$ of $\mathrm{EADP}_{\mathfrak{C}_d^2}(G, \widetilde{E}, \gamma)$:

$$\hat{d}_\alpha = \begin{cases} 1, \text{ if } d_\alpha^* \geq 1/2 \\ 0, \text{ otherwise} \end{cases} \quad E' = \{\alpha \mid \hat{d}_\alpha = 1\} \quad E_1 = \{e \mid (u_e, t) \in E'\} \quad (6)'$$

Lemma 6 E' *is indeed a s-t cut of* $\mathcal{G}$ *and* E' *does not contain any cy-ed-arc.*

Proof. Since the capacity of any cy-ed-arc α in ∞, $d_\alpha^* = 0$ and therefore $\alpha \notin E'$. To see that E' is indeed a s-t cut, consider any s-t path $(s, u_{\mathcal{F}_j}), (u_{\mathcal{F}_j}, u_e)$, (u_e, t). Since $d_{(u_{\mathcal{F}_j}, u_e)}^* =$, we have $d_{(s, u_{\mathcal{F}_j})} + d_{(u_{\mathcal{F}_j}, u_e)} + d_{(u_e, t)} = d_{(s, u_{\mathcal{F}_j})} + d_{(u_e, t)} \geq 1$, which implies $\max\{d_{(s, u_{\mathcal{F}_j})}, d_{(u_e, t)}\} \geq 1/2$, putting at least one edge of the path in E' for deletion.   ❑

Note that $|E_1| = |\{e \mid d_{(u_e, t)}^* \geq 1/2\}| \leq 2 \sum_{e \in \widetilde{E}} d_{(u_e, t)}^* = 2\kappa$, as desired. Let $(\mathcal{S}, \mathcal{V} \setminus \mathcal{S})$ be the s-t cut such that $\mathrm{cut}(\mathcal{S}, \mathcal{V} \setminus \mathcal{S}) = E'$. It thus follows that

$$\mathrm{cap}(\mathrm{cut}(\mathcal{S}, \mathcal{V} \setminus \mathcal{S})) = |E'| = |\{\alpha \mid d_\alpha^* \geq 1/2\}|$$
$$\leq 2 \sum_{\alpha \in \mathcal{E}} \mathrm{cap}(\alpha) d_\alpha^* = 2\,\mathrm{OPT}_{(5)'} \leq 2\,\Gamma \quad (8)$$

Let $\Psi = \{\mathcal{F}_j \mid u_{\mathcal{F}_j} \in \mathcal{S}\}$ and let Υ be the cycles from $\mathcal{C}'(G)$ that are removed by deletion of the edges in E_1. Since *no* cy-ed-arc (of infinite capacity) can be an edge of the minimum s-t cut $(\mathcal{S}, \mathcal{V} \setminus \mathcal{S})$, Ψ is a subset of Υ. The claimed bound on $\mathfrak{C}_d^2(G \setminus E_1)$ can now be shown as follows using (8):

$$\mathrm{cap}(\mathrm{cut}(\mathcal{S}, \mathcal{V} \setminus \mathcal{S}))$$
$$= |\{u_{\mathcal{F}_j} \mid u_{\mathcal{F}_j} \notin \mathcal{S}\}| + |\{u_e \mid u_e \in \mathcal{S}\}|$$
$$= (\Delta' - |\Psi|) + |E_1| \leq 2\,\Gamma$$
$$\Rightarrow (\Delta' - |\Upsilon|) + |E_1| \leq 2\,\Gamma = 2\,\gamma - 2\,\mathfrak{C}_d^2(G) + 2\,\Delta'$$
$$\Rightarrow \mathfrak{C}_d^2(G \setminus E_1) - \mathfrak{C}_d^2(G) = |E_1| - |\Upsilon| \leq 2\,\gamma - 2\,\mathfrak{C}_d^2(G) + \Delta'$$
$$\Rightarrow \frac{\mathfrak{C}_d^2(G \setminus E_1) - \mathfrak{C}_d^2(G)}{\gamma - \mathfrak{C}_d^2(G)} \leq 2 + \frac{\Delta'}{\gamma - \mathfrak{C}_d^2(G)} \leq 1 - \varepsilon$$

(b3) In the proof of Theorem 3, set $\widetilde{E} = E_2 \setminus E_1$ and $\gamma = \mathfrak{C}_d^2(G_2)$. Note that the proof shows that $\mathfrak{C}_d^2(G_2) < \mathfrak{C}_d^2(G_1) <$ The proof also shows that $\mathfrak{C}_d^2(G_1 \setminus E_3) \geq \gamma$ for any proper subset of edges $E_3 \subset \widetilde{E}$, which ensures that for any subset of edges $E_4 \subseteq \widetilde{E}$ $\mathfrak{C}_d^2(G_1 \setminus E_4) \leq \gamma$ is equivalent to stating $\mathfrak{C}_d^2(G_1 \setminus E_4) = \gamma$.

6.2 Gromov-hyperbolic curvature: computational complexity of $\mathrm{EADP}_{\mathfrak{C}_{\mathrm{Gromov}}}$

Theorem 2 *The following statements hold for* $\mathrm{EADP}_{\mathfrak{C}_{\mathrm{Gromov}}}(G, \widetilde{E}, \gamma)$ *when* $\gamma > \mathfrak{C}_{\mathrm{Gromov}}(G)$:

(a) *Deciding if there exists a feasible solution is* NP-*hard.*

(b) *Even if a trivial feasible solution exists, it is* NP*-hard to design a polynomial-time algorithm to approximate* $\mathsf{OPT}_{\mathrm{EADP}_{\mathfrak{C}_{\mathrm{Gromov}}}}(G, \widetilde{E}, \gamma)$ *within a factor of* $c\,n$ *for some constants* $c > 0$, *where* n *is the number of nodes in* G.

6.2.1 Proof of Theorem 2

From a high level point of view, Theorem 2 is proved by suitably modifying the reductions used in the proof of Theorem 4.

To prove (a) we will use a simpler version of the proof of Theorem 4 reusing the same notations. Our graph G will be the same as the graph G_1 in that proof, except that we do *not* add the complete graph $K_{|V''|}$ on the nodes $w_0, w_1, \ldots, w_{|V''|-1}$ and consequently we also do *not* have the edge $\{u, w_0\}$. We set $\widetilde{E} = E'$ and $\gamma = \frac{n}{2}+1$. The proof of Theorem 4 shows that $\mathfrak{C}_{\mathrm{Gromov}}(G) < \gamma$, $\mathfrak{C}_{\mathrm{Gromov}}(G \setminus E') \leq \gamma$ for any subset of edges $E' \subseteq \widetilde{E}$, and $\mathfrak{C}_{\mathrm{Gromov}}(G \setminus E') = \gamma$ for a subset of edges $\emptyset \subset E' \subset \widetilde{E}$ if and only if the given cubic graph has a Hamiltonian path between the two specified nodes, thereby showing NP-hardness of the feasibility problem.

To prove (b) the same construction in the proof of Theorem 4 works: G is the same as the graph G_1 in that proof, $\gamma = \frac{n}{2} + 1$, $\widetilde{E}$ is the set of edges whose deletion produced G_2 from G_1, and the trivial feasible solution is G_2. Note that the proof of Theorem 4 shows $\mathfrak{C}_{\mathrm{Gromov}}(G) < \gamma$, $\mathfrak{C}_{\mathrm{Gromov}}(G \setminus E') \leq \gamma$ for any subset of edges $\emptyset \subset E' \subseteq \widetilde{E}$ and $\mathfrak{C}_{\mathrm{Gromov}}(G_2) = \gamma$.

7 Computational complexity of targeted anomaly detection problems

7.1 Geometric curvatures: computational hardness of $\mathrm{TADP}_{\mathfrak{C}_d^2}(G_1, G_2)$

For two functions $f(n)$ and $g(n)$ of n, we say $f(n) = O^*(g(n))$ if $f(n) = O(g(n)\,n^c)$ for some positive constant c. In the sequel we will use the following two complexity-theoretic assumptions: the *unique games conjecture* (UGC) [52, 71], and the *exponential time hypothesis* (ETH) [44, 45, 78].

Theorem 3
(a) *Computing* $\mathsf{OPT}_{\mathrm{TADP}_{\mathfrak{C}_3^2}}(G_1, G_2)$ *is* NP*-hard.*
(b) *There are no algorithms of the following type for* $\mathrm{TADP}_{\mathfrak{C}_d^2}(G_1, G_2)$ *for* $4 \leq d \leq o(n)$ *when* G_1 *and* G_2 *are* n*-node graphs:*

($b1$) *a polynomial time* $(2 - \varepsilon)$*-approximation algorithm for any constant* $\varepsilon > 0$ *assuming* UGC *is true,*
($b2$) *a polynomial time* $(10\sqrt{5} - 21 - \varepsilon) \approx 1.36$*-approximation algorithm for any constant* $\varepsilon > 0$ *assuming* P$\neq$NP,
($b3$) *a* $O^*\!\left(2^{o(n)}\right)$*-time exact computation algorithm assuming* ETH *is true, and*

($b4$) a $O^*\big(n^{o(\kappa)}\big)$-time exact computation algorithm if $\mathsf{OPT}_{\mathrm{TADP}_{\mathfrak{C}_3^2}}(G_1, G_2) \leq \kappa$ assuming ETH is true.

7.1.1 Proof techniques and relevant comments regarding Theorem 3

(on proof of (a)) We prove the results by reducing the triangle deletion problem (TDP) to that of solving $\mathrm{TADP}_{\mathfrak{C}_3^2}$. TDP was shown to be NP-hard by Yannakakis in [81].

(on proof of (b)) We provide suitable approximation-preserving reductions from MNC.

(on proofs of ($b3$) and ($b4$)) For these proofs, the idea is to start with an instance of 3-SAT, use "sparsification lemma" in [45] to generate a family of Boolean formulae, reduce each of these formula to MNC, and finally reduce each such instance of MNC to a corresponding instance of $\mathrm{TADP}_{\mathfrak{C}_d^2}$.

7.1.2 Proof of Theorem 3

The *minimum node cover* problem (MNC) is defined as follows.

Definition 5 (minimum node cover problem (MNC)) Given a graph G, select a subset of nodes of *minimum* cardinality such that at least one endpoint of *every* edge has been selected.

Let $\mathsf{OPT}_{\mathrm{MNC}}(G)$ denote the cardinality of the subset of nodes that is an optimal solution of MNC. The (standard) Boolean satisfiability problem is denoted by SAT, and its restricted case when every clause has exactly k literals will be denoted by k-SAT [37]. Consider SAT or k-SAT and let Φ be an input instance (*i.e.*, a Boolean formula in conjunctive normal form) of it. The following inapproximability results are known for MNC:

($\star$ $_{\mathrm{MNC}}$) There exists a polynomial time algorithm that transforms a given instance Φ of SAT to an input instance graph $G = (V, E)$ of MNC such that the following holds for any constant $0 < \varepsilon < \frac{1}{4}$, assuming UGC to be true [53]:

$$\text{if } \Phi \text{ is satisfiable then } \mathsf{OPT}_{\mathrm{MNC}}(G) \leq \left(\tfrac{1}{2} + \varepsilon\right)|V|$$
$$\text{if } \Phi \text{ is not satisfiable then } \mathsf{OPT}_{\mathrm{MNC}}(G) \geq (1 - \varepsilon)|V|$$

($\star\star$ $_{\mathrm{MNC}}$) There exists a polynomial time algorithm that transforms a given instance Φ of SAT to an input instance graph $G = (V, E)$ of MNC such that the following holds for any constant $0 < \varepsilon < 16 - 8\sqrt{5}$ and for some $0 < \alpha < 2|V|$, assuming P$\neq$NP [30]:

$$\text{if } \Phi \text{ is satisfiable then } \mathsf{OPT}_{\mathrm{MNC}}(G) \leq \left(\tfrac{\sqrt{5}-1}{2} + \varepsilon\right)\alpha$$
$$\text{if } \Phi \text{ is not satisfiable then } \mathsf{OPT}_{\mathrm{MNC}}(G) \geq \left(\tfrac{71-31\sqrt{5}}{2} - \varepsilon\right)\alpha$$

(note that $\left(\frac{71-31\sqrt{5}}{2}\right) / \left(\frac{\sqrt{5}-1}{2}\right) = 10\sqrt{5} - 21 \approx 1.36$).

($\star\star\star$ $_{\mathrm{MNC}}$) There exists a polynomial time algorithm (*e.g.*, see [37, page 54]) that transforms a given instance Φ of 3-SAT of n variable and m clauses to to an input instance graph $G = (V, E)$ of MNC with $|V| = 3n + 2m$ nodes and $|E| = n + m$ edges such that such that Φ is satisfiable if and only if $\mathsf{OPT}_{\mathrm{MNC}}(G) = n + 2m$.

Proof of (a) We will prove the results by reducing the triangle deletion problem to that of computing $\mathrm{TADP}_{\mathfrak{C}_3^2}$. The *triangle deletion problem* (TDP) can be stated as follows: *Given G find the minimum number of edges (which we will denote by $\mathsf{OPT}_{\mathrm{TDP}}(G)$) to be deleted from G to make it triangle-free.* TDP was shown to be NP-hard by Yannakakis in [81].

Consider an instance $G = (V, E)$ of TDP where $V = \{u_1, \dots, u_n\}$ and $E = \{e_1, \dots, e_m\}$. We create an instance $G_1 = (V', E_1)$ and $G_2 = (V', E_2)$ (with $\emptyset \subset E_2 \subset E_1$) of $\mathrm{TADP}_{\mathfrak{C}_3^2}$ in the following manner:

▷ For each $u_i \in V$, we create a node $v_i \in V'$. There are n such nodes in V'.
▷ If $\{u_i, u_j\} \in E$, then we add the edge $\{v_i, v_j\}$ to E_1. We call these edges as "original" edges. Let E_d be the set of all original edges; note that $|E_d| = m$.
▷ To ensure that G_2 is a connected graph, we add two new nodes w_i^1, w_i^2 in V' corresponding to each node $v_i \in V'$ for $i = 1, 2, \dots, n - 1$, and add three new edges $\{v_i, w_i^1\}$, $\{w_i^1, w_i^2\}$ and $\{w_i^2, v_{i+1}\}$ in E_1. This step adds $2n - 2$ new nodes and $3n - 3$ new edges to V_1 and E_1, respectively. We call the new edges added in this step as "connectivity" edges.
▷ For each $\{u_i, u_j\}) \in E$, we create a new node $v_{i,j}$ in V' and add two new edges $\{u_i, v_{ij}\}$ and $\{v_{ij}, u_j\}$ to E_1. This step creates a new triangle corresponding to each original edge. We call the new edges added in this step as "triangle-creation" edges. This step adds m new nodes and $2m$ new edges to V_1 and E_1, respectively, and exactly m new triangles.

Define $E_2 = E_1 \setminus E_d$. Thus, we have $|V'| = 3n + m - 2$, $|E_1| = 3n + 3m - 3$, $|E_2| = 3n + 2m - 3$, and G_2 contains *no* triangles. Let Δ is the number of triangles in G_1 created using only original edges (the "original triangles"); note that Δ is also equal to the number of triangles in G. Then, $\mathfrak{C}_3^2(G_1) = |V'| - (3m + 3n - 3) + (\Delta + m)$ and $\mathfrak{C}_3^2(G_2) = |V'| - (2m + 3n - 3)$. The following lemma completes our NP-hardness proof.

Lemma 7 $\mathsf{OPT}_{\mathrm{TDP}}(G) = \mathsf{OPT}_{\mathrm{TADP}_{\mathfrak{C}_3^2}}(G_1, G_2)$.

Proof.

Proof of $\mathsf{OPT}_{\mathrm{TDP}}(G) \geq \mathsf{OPT}_{\mathrm{TADP}_{\mathfrak{C}_3^2}}(G_1, G_2)$.

Let $E_{\mathrm{opt}} \subset E$ be an optimum solution of TDP on G, let $E'_{\mathrm{opt}} = \{\{v_i, v_j\} \mid \{u_i, u_j\} \in E\} \subseteq E_d$, and consider the graph $G_3 =$

$(V', E_1 \backslash E'_{\mathrm{opt}})$. Note that G_3 has *no* original triangles and has exactly $m - |E'_{\mathrm{opt}}|$ triangles involving triangle-creation edges, and thus

$$\mathfrak{C}_3^2(G_3) = |V'| - (3n + 3m - 3 - |E'_{\mathrm{opt}}|) + (m - |E'_{\mathrm{opt}}|) = \mathfrak{C}_3^2(G_2)$$

and therefore $\mathsf{OPT}_{\mathrm{TADP}_{\mathfrak{C}_3^2}}(G_1, G_2) \leq |E'_{\mathrm{opt}}| = |E_{\mathrm{opt}}| = \mathsf{OPT}_{\mathrm{TDP}}(G)$.

Proof of $\mathsf{OPT}_{\mathrm{TDP}}(G) \leq \mathsf{OPT}_{\mathrm{TADP}_{\mathfrak{C}_3^2}}(G_1, G_2)$.

Suppose that $E'_{\mathrm{opt}} \subset E_d$ is an optimum solution of q edges of $\mathrm{TADP}_{\mathfrak{C}_3^2}$ on G_1 and G_2, let $G_3 = (V', E_1 \backslash E'_{\mathrm{opt}})$ be the graph obtained from G_1 by removing the edges in E'_{opt}, and let $E' = \{\{u_i, u_j\} \mid \{v_i, v_j\} \in E'_{\mathrm{opt}}\} \subseteq E$. Let $q = |E'_{\mathrm{opt}}|$, $e'_1, e'_2, \ldots, e'_q$ be an arbitrary ordering of the edges in E'_{opt} and δ'_i (for $i = 1, 2, \ldots, q$) is the number of triangles in G_1 that contains the edge e'_i but *none* of the edges $e'_1, \ldots, e'_{i-1}$. Note that, for each i, exactly $\delta'_i - 1$ triangles out of the δ'_i triangles are original triangles. Let $\Delta' \leq \Delta$ be the number of original triangles removed by removing the edges in E'_{opt}; thus, $\Delta' = \sum_{i=1}^{q} (\delta'_i - 1)$. Simple calculations now show that

$$\mathfrak{C}_3^2(G_3) = |V'| - (3n + 3m - 3 - |E'_{\mathrm{opt}}|) + \left(\Delta + m - \sum_{i=1}^{q} \delta_i\right)$$

$$= |V'| - (3n + 3m - 3 - |E'_{\mathrm{opt}}|) + \left(\Delta + m - q - \sum_{i=1}^{q} (\delta_i - 1)\right)$$

$$= |V'| - (3n + 3m - 3 - |E'_{\mathrm{opt}}|) + \left(\Delta + m - |E'_{\mathrm{opt}}| - \Delta'\right)$$

$$= |V'| - (3n + 2m - 3) + (\Delta - \Delta')$$

Consequently, $\mathfrak{C}_3^2(G_3) = \mathfrak{C}_3^2(G_2)$ implies $\Delta' = \Delta$ and E' is a valid solution of TDP on G. This implies $\mathsf{OPT}_{\mathrm{TDP}}(G) \leq |E'| = |E'_{\mathrm{opt}}| = \mathsf{OPT}_{\mathrm{TADP}_{\mathfrak{C}_3^2}}(G_1, G_2)$. ❑

Proofs of (*b1*) and (*b2*)

Consider an instance graph $G = (V, E)$ of MNC with n nodes and m edges where $V = \{v_1, v_2, \ldots, v_n\}$ and $E = \{e_1, e_2, \ldots, e_m\}$. Let $\emptyset \subset V_{\mathrm{MNC}} \subset V$ be an optimal solution of $\mathsf{OPT}_{\mathrm{MNC}}(G) = |V_{\mathrm{MNC}}|$ nodes for this instance of MNC. We then create an instance $G_1 = (V', E_1)$ and $G_2 = (V', E_2)$ (with $\emptyset \subset E_2 \subset E_1$) of $\mathrm{TADP}_{\mathfrak{C}_d^2}$ for a given $d \geq 4$ in the following manner:

- For each $v_i \in V$, we create d new nodes $\{v_i^1, v_i^2, \ldots, v_i^d\}$ in V', and a d-cycle containing the edges $\{v_i^1, v_i^2\}, \{v_i^2, v_i^3\}, \ldots, \{v_i^{d-1}, v_i^d\}, \{v_i^d, v_i^1\}$ in E_1. We call the cycles generated in this step as the "node cycles". This creates a total of dn nodes in V' and dn edges in E_1.
- For each edge $\{v_i, v_j\} \in E$, we do the following:
  - Create $d - 4$ new nodes $u_{i,j,1}^1, u_{i,j,2}^1, \ldots, u_{i,j,\lceil \frac{d-4}{2} \rceil}^1$ and $u_{i,j,1}^2, u_{i,j,2}^2, \ldots, u_{i,j,\lfloor \frac{d-4}{2} \rfloor}^2$ in V'.

– Add $\left\lceil \frac{d-2}{2} \right\rceil$ new edges
   $\{v_i^1, u_{i,j,1}^1\}, \{u_{i,j,1}^1, u_{i,j,2}^1\}, \ldots, \{u_{i,j,\left\lceil \frac{d-4}{2} \right\rceil - 1}^1, u_{i,j,\left\lceil \frac{d-4}{2} \right\rceil}^1\}, \{u_{i,j,\left\lceil \frac{d-4}{2} \right\rceil}^1, v_j^1\}$
   and $\left\lfloor \frac{d-2}{2} \right\rfloor$ new edges
   $\{v_i^2, u_{i,j,1}^2\}, \{u_{i,j,1}^2, u_{i,j,2}^2\}, \ldots, \{u_{i,j,\left\lfloor \frac{d-4}{2} \right\rfloor - 1}^2, u_{i,j,\left\lfloor \frac{d-4}{2} \right\rfloor}^2\}, \{u_{i,j,\left\lfloor \frac{d-4}{2} \right\rfloor}^2, v_j^2\}$
   in E_1. Note that these edges create a d-cycle involving the two edges
   $\{v_i^1, v_i^2\}$ and $\{v_j^1, v_j^2\}$; we refer to this cycle as an "edge cycle".
 These steps create a total of $(d-4)m$ additional nodes in V' and $(d-2)m$
 additional edges in E_1.
– Let $E_2 = E_1 \setminus \{\{v_i^1, v_i^2\} \mid 1 \le i \le n\}$.

Thus, $|V'| = dn + (d-4)m$, $|E_1| = dn + (d-2)m$ and $|E_2| = (d-1)n + (d-2)m$.
To verify that the reduction is possible for any d in the range of values as
claimed in the theorem, note that

$$d \le o(|V'|) \equiv d/|V'| \le o(1) \Leftarrow n^{-1} \le o(1)$$

and the last inequality is trivially true. By ($\star$ $_{\mathrm{MNC}}$) and ($\star\star$ $_{\mathrm{MNC}}$), the
proof is complete once we prove the following lemma.

Lemma 8 $\mathsf{OPT}_{\mathrm{MNC}}(G) = \mathsf{OPT}_{\mathrm{TADP}_{\mathfrak{C}_d^2}}(G_1, G_2)$.

Proof. Let $E_d = E_1 \setminus E_2$. Let f be the total number of cycles of at most d
edges in G_1; thus

$$\mathfrak{C}_d^2(G_1) = |V'| - |E_1| + f = -2m + f$$

Note that any cycle of at most d edges containing an edge from E_d *must* be
either a node cycle or an edge cycle since a cycle containing an edge from E_d
that is *neither* a node cycle *nor* an edge cycle has a number of edges that is
at least $2 + 2 \times \left\lfloor \frac{d-2}{2} \right\rfloor + \left\lceil \frac{d-2}{2} \right\rceil = d + \left\lfloor \frac{d-2}{2} \right\rfloor > d$ since $d \ge 4$. Since removing
all the edges in E_d removes *every* node and *every* edge cycle,

$$\mathfrak{C}_d^2(G_2) = |V'| - |E_2| + (f - n - m) = \left(|V'| - |E_1| + f \right) - m = \mathfrak{C}_d^2(G_1) - m$$

Given an optimal solution $V_{\mathrm{MNC}} \subset V$ of MNC on G of $\mathsf{OPT}_{\mathrm{MNC}}(G)$ nodes,
consider the graph $G_3 = (V', E_3)$ where $E_3 = E_1 \setminus E'_d$ and $E'_d = \{\{v_i^1, v_i^2\} \mid v_i \in V_{\mathrm{MNC}}\} \subseteq E_d$. Since every edge of G is incident on one or more nodes in V_{MNC},
every edge cycle and *exactly* $|E'_d| = \mathsf{OPT}_{\mathrm{MNC}}(G)$ node cycles of G_1 are removed
in G_3, and thus

$$\mathfrak{C}_d^2(G_3) = |V'| - (|E_1| - \mathsf{OPT}_{\mathrm{MNC}}(G)) + (f - \mathsf{OPT}_{\mathrm{MNC}}(G) - m)$$
$$= \mathfrak{C}_d^2(G_1) - m = \mathfrak{C}_d^2(G_2)$$

This shows that $\mathsf{OPT}_{\mathrm{TADP}_{\mathfrak{C}_d^2}}(G_1, G_2) \le \mathsf{OPT}_{\mathrm{MNC}}(G)$. Conversely, consider an
optimal solution $E'_d \subseteq E_d$ of $\mathrm{TADP}_{\mathfrak{C}_d^2}$ for G_1 and G_2, and let $G_3 = (V', E_3)$
where $E_3 = E_1 \setminus E'_d$. Note that *exactly* $|E'_d| = \mathsf{OPT}_{\mathrm{TADP}_{\mathfrak{C}_d^2}}(G_1, G_2)$ node

cycles of G_1 are removed in G_3. Let m' be the number of edge cycles of G_1 removed in G_3. Then,

$$\mathfrak{C}_d^2(G_3) = |V'| - (|E_1| - |E_d'|) + (f - |E_d'| - m') = \mathfrak{C}_d^2(G_1) - m'$$

and consequently m' must be equal to m to satisfy the constraint $\mathfrak{C}_d^2(G_3) = \mathfrak{C}_d^2(G_1) - m$, which implies that G_3 contains *no* edge cycles. This implies that, for every edge cycle involving the two edges $\{v_i^1, v_i^2\}$ and $\{v_j^1, v_j^2\}$ in G_1, at least one of these two edges must be in E_d'', which in turn implies that the set of nodes $V'' = \{v_i \,|\, \{v_i^1, v_i^2\} \in E_d'\}$ in G contains at least one of the nodes v_i or v_j for every edge $\{v_i, v_j\} \in E$. Thus, V'' is a valid solution of MNC on G and $\mathsf{OPT}_{\mathrm{MNC}}(G) \leq |V''| = |E_d''| = \mathsf{OPT}_{\mathrm{TADP}_{\mathfrak{C}_d^2}}(G_1, G_2)$.   ❑

Proof of (*b3*)

We describe the proof for $d = 4$ only; the proof for $d > 4$ is very similar. Suppose, for the sake of contradiction, that one *can* in fact compute $\mathsf{OPT}_{\mathrm{TADP}_{\mathfrak{C}_4^2}}(G_1, G_2)$ in $O^*\big(2^{o(n)}\big)$ time where each of G_1 and G_2 has n nodes. We start with an instance Φ of 3-SAT having n variables and m clauses. The "sparsification lemma" in [45] proves the following result:

> *for every constant $\varepsilon > 0$, there is a constant $c > 0$ such that there exists a $O\big(2^{\varepsilon n}\big)$-time algorithm that produces from Φ a set of t instances $\Phi_1, \ldots, \Phi_t$ of 3-SAT on these n variables with the following properties:*
> - $t \leq 2^{\varepsilon n}$,
> - *each Φ_j is an instance of 3-SAT with $n_j \leq n$ variables and $m_j \leq cn$ clauses, and*
> - *Φ is satisfiable if and only if at least one of $\Phi_1, \ldots, \Phi_t$ is satisfiable.*

For each such above-produced 3-SAT instance Φ_j, we now use the reduction mentioned in $(\star\star\star_{\mathrm{MNC}})$ to produce an instance $G_j = (V_j, E_j)$ of MNC of $|V_j| = 3n_j + 2m_j \leq (3 + 2c)\,n$ nodes and $|E_j| = n_j + m_j \leq (1 + c)\,n$ edges such that Φ_j is satisfiable if and only if $\mathsf{OPT}_{\mathrm{MNC}}(G_j) = n_j + 2m_j$. Now, using the reduction as described in the proof of parts (*b1*) and (*b2*) of this theorem and Lemma 8 thereof, we obtain an instance $G_{1,j} = (V_j', E_{1,j})$ and $G_{2,j} = (V_j', E_{2,j})$ of $\mathrm{TADP}_{\mathfrak{C}_3^2}$ such that $|V_j'| = 4|V_j| < (12 + 8c)n$. By assumption, we can compute $\mathsf{OPT}_{\mathrm{TADP}_{\mathfrak{C}_3^2}}(G_{1,j}, G_{2,j})$ in $O^*\big(2^{o(n)}\big)$, and consequently $\mathsf{OPT}_{\mathrm{MNC}}(G_j)$ in $O^*\big(2^{o(n)}\big)$ time, which in turn leads us to decide in $O^*\big(2^{o(n)}\big)$ time if Φ_j is satisfiable for every j. Since $t \leq 2^{\varepsilon n}$ for every constant $\varepsilon > 0$, this provides a $O^*\big(2^{o(n)}\big)$-time algorithm for 3-SAT, contradicting ETH.

Proof of (*b4*)

The proof is very similar to that in (**b3**) except that now we start with the following lower bound result on parameterized complexity (*e.g.*, see [24, Theorem 14.21]):

> *assuming ETH to be true, if $\mathsf{OPT}_{\mathrm{MNC}}(G) \leq k$ then there is no $O^*\big(n^{o(k)}\big)$-time algorithm for exactly computing $\mathsf{OPT}_{\mathrm{MNC}}(G)$.*

7.2 Gromov-hyperbolic curvature: computational hardness of $\textsc{Tadp}_{\mathfrak{C}_{\text{Gromov}}}$

Theorem 4 *It is* NP-*hard to design a polynomial-time algorithm to approximate* $\textsc{Tadp}_{\mathfrak{C}_{\text{Gromov}}}(G_1, G_2)$ *within a factor of* $c\,n$ *for some constant* $c > 0$, *where* n *is the number of nodes in* G_1 *or* G_2.

7.2.1 Proof techniques and relevant comments regarding Theorem 4

The reduction is from the *Hamiltonian path problem for cubic graphs* (Cubic-Hp), and shown schematically in Fig. 4. Conceptually, the idea is to amplify the difference between Hamiltonian and non-Hamiltonian paths to a large size difference of "geodesic" triangles (*cf.* Definition 2) such that application of results such as [64, Lemma 2.1] can lead to a large difference of the corresponding Gromov-hyperbolicity values. To get the maximum possible amplification (maximum gap in lower bound) we need to make very careful and precise arguments regarding the Gromov-hyperbolicities of classes of graphs. Readers should note that Gromov-hyperbolicity value is not necessarily related to the circumference of a graph, and thus the reduction cannot rely simply on presence or absence of long paths or long cycles in the constructed graph.

   The inapproximability reduction necessarily requires some nodes with large (close to linear) degrees even though with start with Cubic-Hp in which every node has degree exactly 3. We conjecture that our large inapproximability bounds do *not* hold when the given graphs have nodes of bounded degree, but have been unable to prove it so far.

7.2.2 Proof of Theorem 4

We will prove our inapproximability result via a reduction from the *Hamiltonian path* problem for cubic graphs (Cubic-Hp) which is defined as follows.

Definition 6 (Hamiltonian path problem for cubic graphs (Cubic-Hp)) Given a cubic (*i.e.*, a 3-regular) graph $G = (V, E)$ and two specified nodes $u, v \in V$, does G contain a Hamiltonian path between u and v, *i.e.*, a path between u and v that visits every node of G exactly once?

   Cubic-Hp is known to be NP-complete [38]. Consider an instance $G = (V, E)$ and $v_1, v_n \in V$ of Cubic-Hp of n nodes and $m = 3n/2$ edges where $V = \{v_1, v_2, \ldots, v_n\}$, $E = \{e_1, e_2, \ldots, e_m\}$ and the goal is to determine if there is a Hamiltonian path between v_1 and v_n (see Fig. 4 (a)). We first introduce three new nodes v_0, v_{n+1} and v_{n+2}, and connect them to the nodes in G by adding three new edges $\{v_0, v_1\}$, $\{v_n, v_{n+1}\}$ and $\{v_{n+1}, v_{n+2}\}$, resulting in the graph $G' = (V', E')$ (see Fig. 4 (b)). It is then trivial to observe the following:

- G has a Hamiltonian path between v_1 and v_n if and only if G' has a Hamiltonian path between v_0 and v_{n+2}.
- If G' does have a Hamiltonian path then such a path must be between the two nodes v_0 and v_{n+2}.
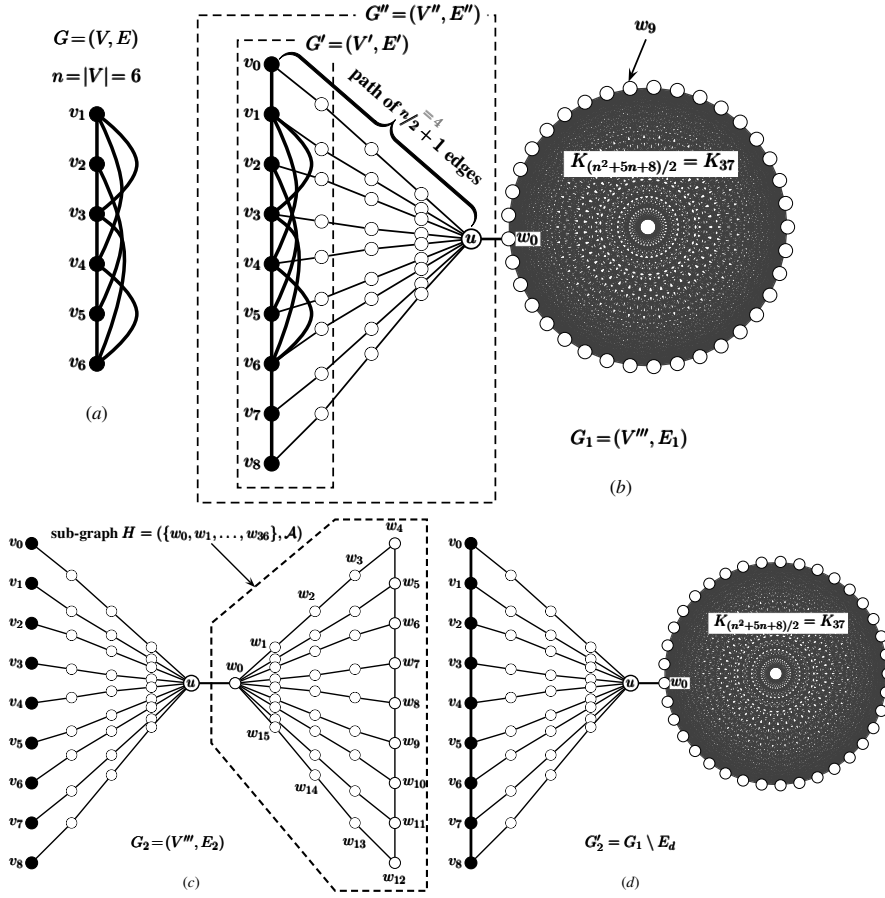
Fig. 4 Illustration of the reduction in Theorem 4. (*a*) The input graph $G = (V, E)$ for the Hamiltonian path problem for cubic graphs (CUBIC-HP). (*b*) and (*c*) The graphs $G_1 = (V'', E_1)$ and $G_2 = (V'', E_2)$ for the generated instance of $\text{TADP}_{\mathfrak{C}_{\text{Gromov}}}(G_1, G_2)$. The graph $G' = (V', E')$ obtained from the given graph G' by adding three extra nodes and three extra edges. (*d*) An optimal solution G_2' for $\text{TADP}_{\mathfrak{C}_{\text{Gromov}}}(G_1, G_2)$ if G contains a Hamiltonian path between v_1 and v_n.

Note that $|V'| = n + 3$ and $|E'| = (3n/2) + 3$. We next create the graph $G'' = (V'', E'')$ from G' in the following manner (see Fig. 4 (*b*)):

- We add a set of $1 + (n^2 + 3n)/2$ new nodes $u, v_{0,1}, \ldots, v_{0,n/2}, v_{1,1}, \ldots, v_{1,n/2}, \ldots, v_{n+2,1}, \ldots, v_{n+2,n/2}$. For notational convenience, we set $u \stackrel{\text{def}}{=} v_{i,0}$ for all $i \in \{0, 1, \ldots, n + 2\}$ and $v_j \stackrel{\text{def}}{=} v_{j,(n/2)+1}$ for all $j \in \{0, 1, \ldots, n + 2\}$.
- We add a set of $n + 3$ disjoint paths (each of length $\frac{n}{2} + 1$) $\mathcal{P}_0, \mathcal{P}_1, \ldots, \mathcal{P}_{n+2}$ where $\mathcal{P}_j \stackrel{\text{def}}{=} v_{j,0} \leftrightarrow v_{j,1} \leftrightarrow v_{j,2} \leftrightarrow \cdots \leftrightarrow v_{j,\frac{n}{2}+1}$.

Note that $|V''| = n+4+\frac{n^2+3n}{2} = \frac{n^2+5n}{2}+4$ and $|E''| = \frac{3n}{2}+3+(n+3)\left(\frac{n}{2}+1\right) = \frac{n^2}{2}+4n+6$. We now create an instance $G_1 = (V, E_1)$ and $G_2 = (V, E_2)$ (with $\emptyset \subset E_2 \subset E_1$) of $\text{TADP}_{\mathfrak{C}_{\text{Gromov}}}(G_1, G_2)$ from G'' in the following manner (see Fig. 4 (b)–(c)):

– The graph $G_1 = (V''', E_1)$ is obtained by modifying G'' as follows:
  – Add a complete graph $K_{|V''|}$ on $|V''| = \frac{n^2+5n}{2}+4$ nodes $w_0, w_1, \ldots, w_{|V''|-1}$ and the edge $\{u, w_0\}$. This step adds $|V''|$ new nodes and $\binom{|V''|}{2}+1$ new edges.

  Thus, we have $|V'''| = 2\,|V''| = n^2 + 5n + 8$, and

  $$|E_1| = |E''| + \binom{|V''|}{2} + 1 = \frac{n^4 + 10n^3 + 43n^2 + 102n + 104}{8}$$

– The graph $G_2 = (V''', E_2)$ is obtained from G_1 as follows. Let $\mathcal{A}$ be the set of edges of a sub-graph of the graph $K_{|V''|}$ (added in the previous step) that is isomorphic to the graph $(V'', \widehat{E})$ where

  $$\widehat{E} = \left(E'' \setminus E\right) \bigcup \left\{\{v_j, v_{j+1}\} \,|\, j \in \{0, 1, \ldots, n+1\}\right\}$$

  and the node w_0 is mapped to the node u in the isomorphism. Such a sub-graph can be trivially found in polynomial time. For notational convenience we number the nodes in this sub-graph such that the order of the nodes in the largest cycle (having $2n+4$ edges) of this sub-graph is $w_0, w_1, \ldots, w_{2n+3}$ (see Fig. 4 (c)). We then set $E_2 = E'' \cup A \cup \{u, w_0\}$. Thus,

  $$|E_2| = |E''| + |\mathcal{A}| + 1 = |E''| + |\widehat{E}| + 1$$
  $$= |E''| + \left(|E''| - |E| + n + 2\right) + 1 = n^2 + \frac{15n}{2} + 12$$

We first need to prove some bounds on the hyperbolicities of various graphs and sub-graphs that appear in our reduction. It is trivial to see that $\mathfrak{C}_{\text{Gromov}}(K_{|V''|}) = 0$. Define $\widetilde{\Delta_{u,v,w}}(G)$ be a geodesic triangle which contributes to the *minimality* of the value of $\mathfrak{C}_{\text{Gromov}}(G)$, *i.e.*, one of the shortest paths, say $\overline{u,v}$, lies in a $\mathfrak{C}_{\text{Gromov}}(\widetilde{\Delta_{u,v,w}}(G))$-neighborhood of the union $\overline{u,w} \cup \overline{v,w}$ of the other two shortest paths, but $\overline{u,v}$ does *not* lie in a δ-neighborhood of $\overline{u,w} \cup \overline{v,w}$ for any $\delta < \mathfrak{C}_{\text{Gromov}}(\widetilde{\Delta_{u,v,w}}(G))$. The following two facts are well-known.

Fact 1 *For any geodesic triangle* $\Delta_{u,v,w}$*, from the definition of* $\mathfrak{C}_{\text{Gromov}}(\Delta_{u,v,w})$ (*cf.* Definition 2) *it follows that*

$$\mathfrak{C}_{\text{Gromov}}(\Delta_{u,v,w}) \leq \max\left\{\left\lfloor \text{dist}_G(u,v)/2\right\rfloor, \left\lfloor \text{dist}_G(v,w)/2\right\rfloor, \left\lfloor \text{dist}_G(u,w)/2\right\rfloor\right\}$$

Fact 2 ([64, Lemma 2.1]) *We may assume that* $\widetilde{\Delta_{u,v,w}}(G)$ *is a* simple *geodesic triangle, i.e., the three shortest paths* $\overline{u,v}$, $\overline{u,w}$ *and* $\overline{v,w}$ *do not share any nodes other than* u*,* v *or* w*.*

Let H denote the (node-induced) sub-graph $(\{w_0, w_1, \ldots, w_{|V''|-1}\}, \mathcal{A})$ of G_2.

Lemma 9 $\mathfrak{C}_{\text{Gromov}}(G_2) = \mathfrak{C}_{\text{Gromov}}(H) = \frac{n}{2} + 1$.

Proof. By Fact 2 $\widetilde{\Delta_{p,q,r}}(G_2)$ must be a simple geodesic triangle and therefore can only include edges in $\mathcal{A}$. Since the diameter of the sub-graph H is $n + 2$, for any geodesic triangle $\Delta_{p,q,r}$ of H we have

$$\max\left\{\lfloor \text{dist}_G(p,q)/2 \rfloor, \lfloor \text{dist}_G(q,r)/2 \rfloor, \lfloor \text{dist}_G(p,r)/2 \rfloor\right\} \leq n + 2$$

and thus by Fact 1 we have $\mathfrak{C}_{\text{Gromov}}(G_2) = \mathfrak{C}_{\text{Gromov}}(H) = \mathfrak{C}_{\text{Gromov}}(\widetilde{\Delta_{p,q,r}}(G_2)) \leq \frac{n}{2} + 1$. Thus, it suffices we provide a simple geodesic triangle $\Delta_{p,q,r}$ of H for some three nodes p, q, r of H such that $\mathfrak{C}_{\text{Gromov}}(\Delta_{p,q,r}(H)) = \frac{n}{2} + 1$. Consider the simple geodesic triangle $\Delta_{w_0, w_{\frac{n}{2}+1}, w_{\frac{3n}{2}+3}}$ of H consisting of the three shortest paths $\mathcal{Q}_1 \overset{\text{def}}{=} w_0 \leftrightarrow w_1 \leftrightarrow w_2 \leftrightarrow \cdots \leftrightarrow w_{\frac{n}{2}} \leftrightarrow w_{\frac{n}{2}+1}$, $\mathcal{Q}_2 \overset{\text{def}}{=} w_{\frac{n}{2}+1} \leftrightarrow w_{\frac{n}{2}+2} \leftrightarrow w_{\frac{n}{2}+3} \leftrightarrow \cdots \leftrightarrow w_{\frac{3n}{2}+2} \leftrightarrow w_{\frac{3n}{2}+3}$ and $\mathcal{Q}_3 \overset{\text{def}}{=} w_{\frac{3n}{2}+3} \leftrightarrow w_{\frac{3n}{2}+4} \leftrightarrow w_{\frac{3n}{2}+5} \leftrightarrow \cdots \leftrightarrow w_{2n+3}, w_0$, and consider the node w_{n+2} that is the mid-point of the shortest path $\mathcal{Q}_2$ (see Fig. 4 (c)). It is easy to verify that the distance of the node w_{n+2} from the union of the two shortest paths $\mathcal{Q}_1$ and $\mathcal{Q}_3$ is $\frac{n}{2} + 1$. ❑

Now, suppose that we can prove the following two claims:

(completeness)   if G has a Hamiltonian path between v_1 and v_n then
$$\mathsf{OPT}_{\text{TADP}_{\mathfrak{C}_{\text{Gromov}}}}(G_1, G_2) \leq \frac{n}{2} + 1$$

(soundness)   if G has no Hamiltonian paths between v_1 and v_n then
$$\mathsf{OPT}_{\text{TADP}_{\mathfrak{C}_{\text{Gromov}}}}(G_1, G_2) \geq \frac{n^3 + 3n^2 + 2n}{2}$$

Note that this proves the theorem since $\frac{\frac{n^3+3n^2+2n}{2}}{\frac{n}{2}+1} > \frac{n^2}{5} = \Omega(|V'''|)$.

Proof of completeness

Suppose that G has a Hamiltonian path between v_1 and v_n, say $v_1 \leftrightarrow v_2 \leftrightarrow v_3 \leftrightarrow \cdots \leftrightarrow v_{n-1} \leftrightarrow v_n$. Thus, G'' has a Hamiltonian path $v_0 \leftrightarrow v_1 \leftrightarrow v_2 \leftrightarrow v_3 \leftrightarrow \cdots \leftrightarrow v_{n-1} \leftrightarrow v_n \leftrightarrow v_{n+1} \leftrightarrow v_{n+2}$ between v_0 and v_{n+2}. We remove the $\frac{n}{2} + 1$ edges in $E_d = E'' \setminus \{\{v_j, v_{j+1}\} \mid j = 0, 1, \ldots, n+1\}$ that are not in this Hamiltonian path resulting in the graph $G_2' = G_1 \setminus E_d$ (see Fig. 4 (d)). To show that $\mathfrak{C}_{\text{Gromov}}(G_2') = \mathfrak{C}_{\text{Gromov}}(G_2)$, note that by Fact 2 $\widetilde{\Delta_{p,q,r}}(G_2')$ must be a simple geodesic triangle and therefore

$$\mathfrak{C}_{\text{Gromov}}(G_2') = \max\left\{\mathfrak{C}_{\text{Gromov}}(G'' \setminus E_d), \mathfrak{C}_{\text{Gromov}}(K_{|V''|})\right\}$$
$$= \max\left\{\mathfrak{C}_{\text{Gromov}}(G'' \setminus E_d), 0\right\} = \mathfrak{C}_{\text{Gromov}}(G'' \setminus E_d)$$

Since $G'' \setminus E_d$ is isomorphic to H, by Lemma 9 $\mathfrak{C}_{\text{Gromov}}(G'' \setminus E_d) = \mathfrak{C}_{\text{Gromov}}(H) = \mathfrak{C}_{\text{Gromov}}(G_2)$.

Proof of soundness

Assume that G has no Hamiltonian paths between v_1 and v_n, and let $E_d \subseteq E_1 \setminus E_2$ be the optimal set of edges that need to be deleted to obtain the graph $G_2' = (V''', E_1 \setminus E_d)$ such that $\mathfrak{C}_{\mathrm{Gromov}}(G_2') = \mathfrak{C}_{\mathrm{Gromov}}(G)$. By Fact 2, $\widetilde{\Delta_{p,q,r}}(G_2')$ must be a simple geodesic triangle and therefore

$$\mathfrak{C}_{\mathrm{Gromov}}(G_2') = \max \left\{ \mathfrak{C}_{\mathrm{Gromov}}(G'' \setminus E_d),\ \mathfrak{C}_{\mathrm{Gromov}}(K_{|V''|} \setminus E_d) \right\}$$
$$= \mathfrak{C}_{\mathrm{Gromov}}(G_2) = \frac{n}{2} + 1 \quad (9)$$

Lemma 10 $\mathfrak{C}_{\mathrm{Gromov}}(G'' \setminus E_d) \leq \frac{n}{2}$.

Proof. Since G has no Hamiltonian paths between v_1 and v_n, $\mathsf{diam}(G' \setminus E_d) \leq n + 1$. Assume, for the sake of contradiction, that $\mathfrak{C}_{\mathrm{Gromov}}(G'' \setminus E_d) \geq \frac{n}{2} + 1$. By Fact 1, we have

$$\mathfrak{C}_{\mathrm{Gromov}}(G'' \setminus E_d) = \mathfrak{C}_{\mathrm{Gromov}}(\widetilde{\Delta_{p,q,r}}(G'' \setminus E_d))$$
$$\leq \max \left\{ \left\lfloor {}^{\mathsf{dist}_{G'' \setminus E_d}(p,q)}\!/\!_2 \right\rfloor,\ \left\lfloor {}^{\mathsf{dist}_{G'' \setminus E_d}(q,r)}\!/\!_2 \right\rfloor,\ \left\lfloor {}^{\mathsf{dist}_{G'' \setminus E_d}(p,r)}\!/\!_2 \right\rfloor \right\}$$

and thus at least one of the three distances in the left-hand-side of the above inequality, say $\mathsf{dist}_{G'' \setminus E_d}(p,q)$, must be at least $n + 2$. Let $\mathcal{L}(\mathcal{C}(H))$ and $\mathcal{L}(H)$ denote the length (number of edges) of a (simple) cycle $\mathcal{C}$ and the length of the *longest (simple) cycle* of a graph H. Since $\mathfrak{C}_{\mathrm{Gromov}}(\widetilde{\Delta_{p,q,r}}(G'' \setminus E_d)) > 0$ and $\widetilde{\Delta_{p,q,r}}(G'' \setminus E_d)$ must be a simple geodesic triangle, there must be at least one cycle, say $\mathcal{C}$, in $G'' \setminus E_d$ containing p, q and r. Now, note that

$$\mathcal{L}(\mathcal{C}(G'' \setminus E_d)) \leq \mathcal{L}(G'' \setminus E_d) \leq 2\left(\frac{n}{2} + 1\right) + \mathsf{diam}(G' \setminus E_d) \leq 2n + 3$$

and therefore $\mathsf{dist}_{G'' \setminus E_d}(p,q) \leq \left\lfloor \frac{2n+3}{2} \right\rfloor = n + 1$, which provides the desired contradiction.   ❑

By Lemma 10 and Equation (9) it follows that $\mathfrak{C}_{\mathrm{Gromov}}(K_{|V''|} \setminus E_d) = \frac{n}{2} + 1$.

Lemma 11 *If* $\mathfrak{C}_{\mathrm{Gromov}}(K_{|V''|} \setminus E_d) \geq \frac{n}{2} + 1$ *then* $|E_d| \geq \frac{n^3 + 3n^2 + 2n}{2}$.

Proof. Since $\mathfrak{C}_{\mathrm{Gromov}}(K_{|V''|} \setminus E_d) = \mathfrak{C}_{\mathrm{Gromov}}(\widetilde{\Delta_{p,q,r}}(K_{|V''|} \setminus E_d)) = \frac{n}{2} + 1$, by Fact 1 at least one of the three distances $\mathsf{dist}_{K_{|V''|} \setminus E_d}(p,q)$, $\mathsf{dist}_{K_{|V''|} \setminus E_d}(q,r)$ or $\mathsf{dist}_{K_{|V''|} \setminus E_d}(p,r)$, say $\mathsf{dist}_{K_{|V''|} \setminus E_d}(p,q)$, must be at least $n+2$. This implies that $K_{|V''|} \setminus E_d$ must contain a shortest path of length $n + 2$, say $\mathcal{Q} \stackrel{\mathrm{def}}{=} w_0 \leftrightarrow w_1 \leftrightarrow w_2 \leftrightarrow \cdots \leftrightarrow w_{n+1} \leftrightarrow w_{n+2}$. We now claim that *no* node from the set $W_1 = \{w_{n+3}, w_{n+4}, \ldots, w_{|V''|-1}\}$ is connected to *more than* 3 nodes from the set $W_2 = \{w_0, w_1, \ldots, w_{n+2}\}$ in $K_{|V''|} \setminus E_d$. To show this by contradiction, suppose that some node $w_i \in W_1$ is connected to four nodes $w_j, w_k, w_\ell, w_r \in W_2$ with $j < k < \ell < r$. Then $r \geq j+3$ which implies $\mathsf{dist}_{K_{|V''|} \setminus E_d}(w_j, w_r) \leq 2$, contradicting the fact that $\mathcal{Q}$ is a shortest path. It thus follows that

$$|E_d| \geq \big((n+3)-3\big)|W_1| = n(|V''| - (n+3))$$
$$= n\left(\frac{n^2+3n}{2}+1\right) = \frac{n^3+3n^2+2n}{2}$$

❏

The above lemma completes the proof of soundness of our reduction.

8 Conclusion and future research

Notions of curvatures of higher-dimensional geometric shapes and topological spaces play a *fundamental* role in physics and mathematics in characterizing anomalous behaviours of these higher dimensional entities. However, using curvature measures to detect anomalies in networks is *not* yet very common due to several reasons such as lack of preferred geometric interpretation of networks and lack of experimental evidences that may lead to specific desired curvature properties. In this paper we have attempted to formulate and analyze curvature analysis methods to provide the foundations of systematic approaches to find critical components and anomaly detection in networks by using two measures of network curvatures, namely the Gromov-hyperbolic curvature and the geometric curvature measure. This paper must *not* be viewed as uttering the final word on appropriateness and suitability of specific curvature measures, but rather should be viewed as a stimulator and motivator of further theoretical or empirical research on the exciting interplay between notions of curvatures from network and non-network domains.

There is a *plethora* of interesting future research questions and directions raised by the topical discussions and results in this paper. Some of these are stated below.

▷ For geometric curvatures, we considered the first-order non-trivial measure $\mathfrak{C}_d^2$. It would be of interest to investigate computational complexity issues of anomaly detection problems using $\mathfrak{C}_d^p$ for $p > 2$. We conjecture that our algorithmic results for extremal anomaly detection using $\mathfrak{C}_d^2$ (Theorem 1(a2-2)&(b2)) can be extended to $\mathfrak{C}_d^3$.

▷ There are at least two more aspects of geometric curvatures that need further careful investigation. Firstly, the topological association of elementary components to higher-dimensional objects as described in this paper is by *no* means the only reasonable topological association possible. But, more importantly, other suitable notions of geometric curvatures are quite possible. As a very simple illustration, assuming that smaller dimensional simplexes edges in the discrete network setting correspond to vectors or directions in the smooth context, an analogue of the Bochner-Weitzenböck formula developed by Forman for the curvature for a simplex

s can be given by the formula [34, 68]:

$$\mathfrak{F}(s) = w_s \left(\left(\sum_{s \prec s'} \frac{w_s}{w_{s'}} + \sum_{s' \prec s} \frac{w_{s'}}{w_s} \right) - \sum_{s' \| s} \Big| \sum_{s, s' \prec g} \frac{\sqrt{w_s w_{s'}}}{w_g} + \sum_{g \prec s, s'} \frac{w_g}{\sqrt{w_s w_{s'}}} \Big| \right)$$

where $a \prec b$ means a is a face of b, $a \parallel b$ means a and b have either a common higher-dimensional face or a common lower-dimensional face but *not* both, and w is a function that assigns weights to simplexes. One can then either modify the Euler characteristics as $\sum_{k=0}^{p} (-1)^k \mathfrak{F}(f_d^k)$ or by combining the individual $\mathfrak{F}(f_d^k)$ values using curvature functions defined by Bloch [12].

▷ Our inapproximability results for the Gromov-hyperbolic curvature require a high average node degree. We hypothesize that the anomaly detection problems using Gromov-hyperbolic curvatures is much more computationally tractable than what our results depict for networks with bounded average degree.

Finally, in contrast to the combinatorial/geometric graph-property based approach investigated in this paper, a viable alternate approach for anomaly detection is the *algebraic tensor-decomposition based approach* studied in the contexts of dynamic social networks [69] and pathway reconstructions in cellular systems and microarray data integration from several sources [6, 61]. This approach is quite different from the ones studied in this paper with its own pros and cons. For computational biology researchers, an useful survey of tensor-based approaches for various kinds of biological networks and systems can be found in reference [79].

Acknowledgements

References

1. R. Albert and A.-L. Barabási. *Statistical mechanics of complex networks*, Reviews of Modern Physics, 74(1), 47-97, 2002.
2. R. Albert, B. DasGupta and N. Mobasheri. *Topological implications of negative curvature for biological and social networks*, Physical Review E, 89(3), 032811, 2014.
3. R. Albert, B. DasGupta, R. Dondi and E. Sontag. *Inferring (Biological) Signal Transduction Networks via Transitive Reductions of Directed Graphs*, Algorithmica, 51(2), 129-159, 2008.
4. R. Albert, B. DasGupta, R. Dondi, S. Kachalo, E. Sontag, A. Zelikovsky and K. Westbrooks. *A Novel Method for Signal Transduction Network Inference from Indirect Experimental Evidence*, Journal of Computational Biology, 14(7), 927-949, 2007.
5. R. Albert, B. DasGupta, A. Gitter, G. Gürsoy, R. Hegde, P. Pal, G. S. Sivanathan and E. D. Sontag. *A New Computationally Efficient Measure of Topological Redundancy of Biological and Social Networks*, Physical Review E, 84(3), 036117, 2011.

6. O. Alter and G. H. Golub. *Reconstructing the pathways of a cellular system from genome-scale signals by using matrix and tensor computations*, PNAS, 102(49), 17559-17564, 2005.

7. S. Aminikhanghahi and D. J. Cook. *A Survey of Methods for Time Series Change Point Detection*, Knowledge and Information Systems, 51(2), 339-367, 2017.

8. F. Ariaei, M. Lou, E. Jonckeere, B. Krishnamachari and M. Zuniga. *Curvature of sensor network: clustering coefficient*, EURASIP Journal on Wireless Communications and Networking, 213185, 2008.

9. D. S. Bassett, N. F. Wymbs, M. A. Porter, P. J. Mucha, J. M. Carlson and S. T. Grafton. *Dynamic reconfiguration of human brain networks during learning*, PNAS, 108(18), 7641-7646, 2011.

10. I. Benjamini. *Expanders are not hyperbolic*, Israel Journal of Mathematics, 108, 33-36, 1998.

11. M. Berger. *A Panoramic View of Riemannian Geometry*, Springer, 2012.

12. E. Bloch. *Combinatorial Ricci Curvature for Polyhedral Surfaces and Posets*, preprint, arXiv:1406.4598v1 [math.CO], 2014.

13. M. Bonk and O. Schramm. *Embeddings of Gromov hyperbolic spaces*, Geometric and Functional Analysis, 10, 266-306, 2000.

14. M. Bosc, F. Heitz, J. P. Armspach, I. Namer, D. Gounot and L. Rumbach. *Automatic change detection in multimodal serial MRI: application to multiple sclerosis lesion evolution*, Neuroimage, 20(2), 643-656, 2003.

15. M. R. Bridson and A. Haefliger. *Metric Spaces of Non-Positive Curvature*, Springer, 1999.

16. J. Chalopin, V. Chepoi, F. F. Dragan, G. Ducoffe, A. Mohammed and Y. Vaxès. *Fast approximation and exact computation of negative curvature parameters of graphs*, to appear in Discrete and Computational Geometry.

17. V. Chepoi, F. F. Dragan, B. Estellon, M. Habib and Y. Vaxès. *Diameters, centers, and approximating trees of δ-hyperbolic geodesic spaces and graphs*, proceedings of the 24th Annual Symposium on Computational geometry, 59-68, 2008.

18. V. Chepoi, F. F. Dragan, B. Estellon, M. Habib, Y. Vaxès and Y. Xiang. *Additive spanners and distance and routing labeling schemes for δ-hyperbolic graphs*, Algorithmica, 62(3-4), 713-732, 2012.

19. V. Chepoi and B. Estellon. *Packing and covering δ-hyperbolic spaces by balls*, in Lecture Notes in Computer Science 4627, M. Charikar, K. Jansen, O. Reingold and J. D. P. Rolim (Eds.), 59-73, Springer, 2007.

20. M. Chlebík and J. Chlebíková. *Complexity of approximating bounded variants of optimization problems*, Theoretical Computer Science, 354, 320-338, 2006.

21. M. F. R. Chowdhury, S. A. Selouani and D. O'Shaughnessy. *Bayesian on-line spectral change point detection: a soft computing approach for on-line ASR*, International Journal of Speech Technology, 15(1), 5-23, 2011.

22. V. Colizza, A. Flammini, M. A. Serrano and A. Vespignani. *Detecting rich-club ordering in complex networks*, Nature Physics, 2, 110-115, 2006.

23. W. J. Cook, W. H. Cunningham, W. R. Pulleyblank and A. Schrijver. *Combinatorial Optimization*, Wiley, 1997.

24. M. Cygan, F. Fomin, L. Kowalik, D. Lokshtanov, D. Marx, M. Pilipczuk, M. Pilipczuk and S. Saurabh. *Parameterized Algorithms*, Springer, 2015.

25. B. DasGupta and D. Desai. *Complexity of Newman's Community Finding Approach for Social Networks*, Journal of Computer & System Sciences, 79, 50-67, 2013.

26. B. DasGupta, M. Karpinski, N. Mobasheri and F. Yahyanejad. *Effect of Gromov-hyperbolicity Parameter on Cuts and Expansions in Graphs and Some Algorithmic Implications*, Algorithmica, 80(2), 772-800, 2018.

27. B. DasGupta and J. Liang. *Models and Algorithms for Biomolecules and Molecular Networks*, John Wiley & Sons, Inc., 2016.

28. R. Duan. *Approximation Algorithms for the Gromov Hyperbolicity of Discrete Metric Spaces*, in A. Pardo and A. Viola (Eds.), LATIN 2014, LNCS 8392, 285-293, 2014.

29. F. de Montgolfier, M. Soto and L. Viennot. *Treewidth and Hyperbolicity of the Internet*, proceedings of the 10th IEEE International Symposium on Networking Computing and Applications, 25-32, 2011.

30. I. Dinur and S. Safra. *On the hardness of approximating minimum vertex cover*, Annals of Mathematics, 162(1), 439-485, 2005.

31. J. F. Ducre-Robitaille, L. A. Vincent and G. Boulet. *Comparison of techniques for detection of discontinuities in temperature series*, International Journal of Climatology, 23(9), 1087-1101, 2003.

32. B. Fiedler, A. Mochizuki, G. Kurosawa and D. Saito. *Dynamics and control at feedback vertex sets. I: Informative and determining nodes in regulatory networks*, Journal of Dynamics and Differential Equations, 25(3), 563-604, 2013.

33. U. Feige and M. Seltser, *On the densest k-subgraph problems*, Technical Report # CS97-16, Faculty of Mathematics and Computer Science, Weizmann Institute of Science, Israel (available online at `https://pdfs.semanticscholar.org/eb26/c37ff9da661fdc55a0cb8d31ab323e901b3b.pdf`).

34. R. Forman. *Bochner's method for cell complexes and combinatorial ricci curvature*, Discrete and Computational Geometry, 29(3), 323-374, 2003.

35. H. Fournier, A. Ismail, and A. Vigneron. *Computing the Gromov hyperbolicity of a discrete metric space*, Information Processing Letters, 115, 6-8, 576-579, 2015.

36. T. W. Gamelin and R. E. Greene. *Introduction to Topology*, Dover publications, 1999.

37. M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W. H. Freeman, 1979.

38. M. R. Garey, D. S. Johnson and R. E. Tarjan. *The planar Hamiltonian circuit problem is* NP-*complete*, SIAM Journal of Computing, 5, 704-714, 1976.

39. C. Gavoille and O. Ly. *Distance labeling in hyperbolic graphs*, in Lecture Notes in Computer Science 3827, X. Deng and D.-Z. Du (Eds.), 1071-1079, Springer, 2005.

40. A. V. Goldberg. *Finding a maximum density subgraph*, Technical report, 1984.

41. M. Gromov. Hyperbolic groups, *Essays in group theory*, 8, 75-263, 1987.

42. S. Hawking and R. Penrose. *The Nature of Space and Time*, Princeton University Press, 1996.

43. M. Henle. *A Combinatorial Introduction to Topology*, Dover publications, 1994.

44. R. Impagliazzo and R. Paturi. *On the Complexity of k-*Sat, Journal of Computer and System Sciences, 62, 367-375, 2001.

45. R. Impagliazzo, R. Paturi and F. Zane. *Which problems have strongly exponential complexity?*, Journal of Computer and System Sciences, 63(4), 512-530, 2001.

46. A. S. Jarrah, R. Laubenbacher, B. Stigler and M. Stillman. *Reverse-engineering polynomial dynamical systems*, Advances in Applied Mathematics, 39 (4) 477-489, 2007.

47. E. A. Jonckheere and P. Lohsoonthorn. *Geometry of network security*, American Control Conference, 2, 976-981, 2004.

48. E. Jonckheere, P. Lohsoonthorn and F. Ariaei. *Scaled Gromov Four-Point Condition for Network Graph Curvature Computation*, Internet Mathematics, 7(3), 137-177, 2011.

49. E. Jonckheere, P. Lohsoonthorn and F. Bonahon. *Scaled Gromov hyperbolic graphs*, Journal of Graph Theory, 57(2), 157-180, 2007.

50. E. Jonckheerea, M. Loua, F. Bonahona and Y. Baryshnikova. *Euclidean versus hyperbolic congestion in idealized versus experimental networks*, Internet Mathematics, 7(1), 1-27, 2011.

51. Y. Kawahara and M. Sugiyama. *Sequential Change-Point Detection Based on Direct Density-Ratio Estimation*, SIAM International Conference on Data Mining, 389-400, 2009.

52. S. Khot. *On the power of unique 2-Prover 1-Round games*, 34th ACM Symposium on Theory of Computing, 767-775, 2002.

53. S. Khot and O. Regev. *Vertex cover might be hard to approximate to within 2-ε*, Journal of Computer and System Sciences, 74(3), 335-349, 2008.

54. B. Kolb and I. Q. Whishaw. *Fundamentals of Human Neuropsychology*, Freeman, New York, 1996.

55. V. Latora and M. Marchior. *A measure of centrality based on network efficiency*, New Journal of Physics, 9, 188, 2007.

56. C. T. Lin. *Structural controllability*, IEEE Transactions on Automatic Control, 19(3), 201-208, 1974.

57. Y. Y. Liu, J. J. Slotine and A. L. Barabási. *Controllability of complex networks*, Nature, 473(7346), 167-173, 2011.

58. A. Mochizuki, B. Fiedler, G. Kurosawa and D. Saito. *Dynamics and control at feedback vertex sets. II: A faithful monitor to determine the diversity of molecular activities in regulatory networks*, Journal of theoretical biology, 335, 130-146, 2013.

59. D. Narayan and I. Saniee. *Large-scale curvature of networks*, Physical Review E, 84, 066108, 2011.

60. M. E. J. Newman. *Networks: An Introduction*, Oxford University Press, 2010.

61. L. Omberg, G. H. Golub and O. Alter. *A tensor higher-order singular value decomposition for integrative analysis of DNA microarray data from different studies*, PNAS, 104(47), 18371-18376, 2007.

62. F. Papadopoulos, D. Krioukov, M. Boguna and A. Vahdat. *Greedy Forwarding in Dynamic Scale-Free Networks Embedded in Hyperbolic Metric Spaces*, IEEE Conference on Computer Communications, 1-9, 2010.

63. J. Reeves, J. Chen, X. L. Wang, R. Lund and Q. Q. Lu. *A Review and Comparison of Changepoint Detection Techniques for Climate Data*, Journal of Applied Meteorological Climatology, 46(6), 900-915, 2007.

64. J. M. Rodríguez and E. Tourís. *Gromov hyperbolicity through decomposition of metric spaces*, Acta Mathematica Hungarica, 103, 53-84, 2004.

65. J. Roe. *Index Theory, Coarse Geometry, and Topology of Manifolds*, Conference Board of the Mathematical Sciences Regional Conference, Series 90, American Mathematical Society, 1996.

66. D. Rybach, C. Gollan, R. Schluter and H. Ney. *Audio segmentation for speech recognition using segment features*, IEEE International Conference on Acoustics, Speech and Signal Processing. 4197-4200, 2009.

67. A. Saadatpour, R. S. Wang, A. Liao, X. Liu, T. P. Loughran, I. Albert and R. Albert. *Dynamical and structural analysis of a T cell survival network identifies novel candidate therapeutic targets for large granular lymphocyte leukemia*, PLoS Computational Biology, 7, e1002267, 2011.

68. A. Samal, R. P. Sreejith, J. Gu, S. Liu, E. Saucan and J. Jost. *Comparative analysis of two discretizations of Ricci curvature for complex networks*, Scientific Reports, 8, Article number: 8650, 2018.

69. J. Sun, D. Tao and C. Faloutsos. *Beyond Streams and Graphs: Dynamic Tensor Analysis*, 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 374-383, 2006.

70. G. Tononi, O. Sporns and G. M. Edelman. *Measures of degeneracy and redundancy in biological networks*, PNAS, 96, 3257-3262, 1999.

71. L. Trevisan. *On Khot's Unique Games Conjecture*, Bulletin of the American Mathematical Society, 49(1), 91-111, 2012.

72. V. Vazirani. *Approximation Algorithms*, Springer, 2001.

73. A. Wagner. *Estimating Coarse Gene Network Structure from Large-Scale Gene Perturbation Data*, Genome Research, 12, 309-315, 2002.

74. S. Wasserman and K. Faust. *Social Network Analysis: Methods and Applications*, Cambridge University Press, 1994.

75. M. Weber, E. Saucan and J. Jost. *Can one see the shape of a network?*, arXiv:1608.07838v2 [math.CO], 2016.

76. M. Weber, J. Jost and E. Saucan. *Forman-Ricci flow for change detection in large dynamic data sets*, International Conference on Information and Computational Science, 2016.

77. V. V. Williams. *Multiplying matrices faster than Coppersmith-Winograd*, 44th ACM Symposium on Theory of Computing, 887-898, 2012.

78. G. Woeginger. *Exact algorithms for NP-hard problems: A survey*, in Combinatorial Optimization – Eureka, You Shrink!, 2570, 185-207, Springer-Verlag, 2003.

79. F. Yahyanejad, B. DasGupta and R. Albert. *A survey of some tensor analysis techniques for biological systems*, to appear in Quantitative Biology.

80. P. Yang, G. Dumont and J. M. Ansermino. *Adaptive change detection in heart rate trend monitoring in anesthetized children*, IEEE Transactions on Biomedical Engineering, 53(11), 2211-2219, 2006.

81. M. Yannakakis. *Node- and Edge-Deletion NP-Complete Problems*, 10th Annual ACM Symposium on Theory of Computing, 253-264, 1978.

82. J. G. T. Zanudo and R. Albert. *Cell Fate Reprogramming by Control of Intracellular Network Dynamics*, PLOS Computational Biology, 11, e1004193, 2015.