On Density-based Local Community Search

YIZHOU DAI, The University of Auckland, New Zealand
MIAO QIAO, The University of Auckland, New Zealand
RONG-HUA LI, Beijing Institute of Technology, China

Given a graph G and a set R of seed nodes, local community search (LCS) reports a community that is local to R. Specifically, for an induced subgraph S of G, the objective function $f(S)$ not only considers classic community measurement of S such as conductance and density, but also encodes set inclusion criteria of R; LCS optimizes $f(S)$ over all the induced subgraphs of G. Ideally, the optimization algorithm for $f(S)$ should be strongly local; that is, its complexity depends on R as opposed to the entire graph G. This paper formulates a general form of objective functions for LCS using configurations — one configuration corresponds to one LCS objective function. For the set C of configurations corresponding to density-based LCS, this paper i) finds $C_L \subseteq C$ in a constructive classification of C: a configuration in C has a strongly local algorithm for optimizing its corresponding objective function if and only if it is in C_L, and ii) provides a linear programming based general solution for density-based LCS — the solution is strongly local and ready to be deployed to practical scenarios.

CCS Concepts: • **Information systems** → *Clustering*; • **Mathematics of computing** → **Graph algorithms**; • **Theory of computation** → *Linear programming*; • **Computing methodologies** → *Optimization algorithms.*

Additional Key Words and Phrases: dense subgraph search, local community detection, weight configuration, strongly local algorithms, general framework

ACM Reference Format:
Yizhou Dai, Miao Qiao, and Rong-Hua Li. 2024. On Density-based Local Community Search. *Proc. ACM Manag. Data* 2, 2 (PODS), Article 88 (May 2024), 25 pages. https://doi.org/10.1145/3651589

1 INTRODUCTION

Graph database is a NoSQL database that models the relations among real-world entities with nodes and edges. To discover the organizations of graph nodes, graph databases (e.g., Neo4j) usually provide libraries for detecting communities for graph analytical tasks such as link predictions and target recommendations. With the growing size of the underlying graphs, such as social networks, finding scalable search algorithms for quality communities has been studied for two decades (see [5] as an entrance). A line of community search [2, 6, 7, 9, 13, 14] falls into the scope of query optimization. Given a graph $G(V, E)$ and a set of nodes $R \subseteq V$, find a local community which is an induced subgraph $S \subseteq V$ that is close to R and optimizes an objective function $f(S)$. The edge set of S is $E(S) = E \cap (S \times S)$ and $f(S)$ usually evaluates S based on a community measurement such as conductance and density. We refer to the above problem as *local community search* (LCS)

Authors' addresses: Yizhou Dai, ydai992@aucklanduni.ac.nz, The University of Auckland, Auckland, New Zealand; Miao Qiao, miao.qiao@auckland.ac.nz, The University of Auckland, Auckland, New Zealand; Rong-Hua Li, lironghuabit@126.com, Beijing Institute of Technology, Beijing, China.

ACM 2836-6573/2024/5-ART88
https://doi.org/10.1145/3651589

with *seed set* R[1]. The local community search in our context is different from that in [3, 8, 10] which does not involve a seed set: their "locality" requires that the targeted subgraph $S \subseteq V$ should be no worse than its subgraphs and better than its supergraphs under a community metric [8, 10].

Local community search requires that the targeted S should be close to R. Such a local constraint can be *hard-coded* [11, 12], i.e, to enforce $R \subseteq S$, or *soft-coded*, i.e., to encode the set inclusion criteria of R into the objective function $f(S)$. This paper focuses on the latter case: the resulting subgraph depends solely on the formulation of $f(S)$ while the usability of the corresponding LCS also depends on the availability of an efficient algorithm for optimizing $f(S)$. To describe an ideal property of the search algorithm of LCS, a group of LCS adopts the notion of "*strong locality*" [2, 9, 13–15]:

> A LCS algorithm is strongly local if its complexity depends only on the seed set and not the input graph.

With a specific $f(S)$ determined by the choice of base community measure (e.g., density and conductance) and the encoding of the set inclusion criteria, a natural question is: *is there a strongly local algorithm for its optimization?* Further, *is there a strongly local algorithm for a class of $f(S)$*?

To this end, we formulate a general LCS objective function $f(S)$ by summarizing the set inclusion criteria to R with a configuration Ω of 10 real numbers. Ω describes how $f(S)$ favors an edge e based on e's 10 possible relations to sets R and S. This general form has existing soft-coded LCS as special cases. We focus on C, a set of configurations that correspond to density-based LCS whose objective function $f(S)$ has density $\rho(S) = \frac{|E(S)|}{|S|}$ as its base community measure. We provide a classification $C_L \subseteq C$ of C based on whether there exists, for their corresponding objective function $f(S)$, a strongly local algorithm. Moreover, for a large portion of configurations in C_L, we provide a linear programming based general solution that is strongly local and practically efficient.

This section is organized as follows. Section 1.1 first explains how existing objective functions encode set inclusion criteria and then introduces our general form of objective function $f(S)$ for density-based LCS. Section 1.2 shows existing strongly local computation for LCS. Section 1.3 introduces our results on density-based LCS. Section 1.4 demonstrates how to use our results in real-world applications. Table 1 overviews existing local community search.

1.1 Encode Seed Set Inclusion in Optimization

To ensure that the resulting graph is close to R, existing encoding of R in the objective function $f(S)$ [4, 9, 13–15] for a set $S \subseteq V$ is usually done by penalizing nodes in (or the edges on) $S \backslash R$ and $R \backslash S$. In other words, $f(S)$ discourages misaligning R and S. Apart from distance-based penalties [4, 11], a majority of node penalties are degree-based, i.e., for a node $v \in S \setminus R$, the penalty charged on v is proportional to its degree $\deg(v)$. Thus, the encoding of R in the objective function, as shown in Table 1, is usually embodied in a term $\mathrm{vol}(S \setminus R)$, the total degree of nodes in $S \setminus R$, charged to the numerator $|E(S)|$ of density $\rho(S) = \frac{|E(S)|}{|S|}$ or the denominator $\mathrm{vol}(S)$ of conductance[2] $\Phi(S) = \frac{\partial S}{\mathrm{vol}(S)}$ (cut ∂S of S is the number of edges between S and $\overline{S}$), which we call the **key term of a community metric**. With penalties, the key term of density-based LCS [2] becomes $2|E(S)| - \mathrm{vol}(S \setminus R)$, and the key term of conductance-based LCS [9, 13–15] becomes $\mathrm{vol}(S) - (1 + \epsilon) \cdot \mathrm{vol}(S \setminus R)$ where ϵ is a parameter[3]. We make the following observation on these penalized key terms.

[1]This problem has various names. It has been called "local graph partitioning" [1], "local community search" [7], "community search" [11], "anchored subgraph search" [2], "cut improvement" [9, 14], and "seed set expansion" [6]; set R has been called "seed set" [9, 14], "query nodes" [11], and anchored node set [2].

[2]Because $\mathrm{vol}(S) + \mathrm{vol}(\overline{S}) = \mathrm{vol}(V)$, we assume that S has $\mathrm{vol}(S) \leq \frac{\mathrm{vol}(V)}{2}$.

[3]One work [15] takes $|R|$ user-defined parameters, $\{p_r, r \in R\}$, to further penalize each node $r \in R \setminus S$ with $p_r \deg(r)$.

Ref.	Parameters	Hard Coded	Community Metric	Opt. Method	Objective Function	Strongly Local	Flow Based	Edge Weighted
[12]	α	Y	Quasi-Clique	max	$f_\alpha(S) = \lvert E(S)\rvert - \alpha\binom{\lvert S\rvert}{2}$	N	N	Y
[11]	$d > 0$	Y	Min-Degree	max	$\min_{v\in S} \deg(v)$, s.t. $\forall v \in S, \forall r \in R, dist(v,r) \leq d$	N	N	N
[4]	α	N	Density	max	$\frac{\lvert E(S)\rvert - \alpha \sum_{v\in S} \min_{r\in R} d(v,r)}{\lvert S\rvert}$	N	N	N
[9]	$\epsilon = \frac{\mathrm{vol}(R)}{\mathrm{vol}(\overline{R})}$, $C \in [\epsilon/3, 1]$	N	Conductance	min	$\frac{\partial S}{\mathrm{vol}(R\cap S) - \epsilon \mathrm{vol}(S\setminus R)}$ s.t. $\frac{\mathrm{vol}(S\cap R)}{\mathrm{vol}(S)} \geq C$	Y	Y	Y
[13, 14]	$\epsilon > \frac{\mathrm{vol}(R)}{\mathrm{vol}(\overline{R})}$	N	Conductance	min	$\frac{\partial S}{\mathrm{vol}(S) - (1+\epsilon)\mathrm{vol}(S\setminus R)}$	Y	Y	Y
[15]	$\epsilon > \frac{\mathrm{vol}(R)}{\mathrm{vol}(\overline{R})}$, $p_r \geq 0, \forall r \in R$	N	Conductance	min	$\frac{\partial S}{\mathrm{vol}(S) - (1+\epsilon)\mathrm{vol}(S\setminus R) - \sum_{r\in R\setminus S} p_r \deg(r)}$	Y	Y	Y
[2]	-	N	Density	max	$\frac{2\lvert E(S)\rvert - \mathrm{vol}(S\setminus R)}{\lvert S\rvert}$	Y	Y	Y
Ours	Ω (See Def. 1)	N	Density	max	$\max_{S\subseteq V} \frac{g_{\Omega,R}(S)}{\lvert S\rvert}$ (See Def. 3)	See Fig. 5	N	Y

Table 1. Overview: Local Community Search with Seed Set R on Graph G

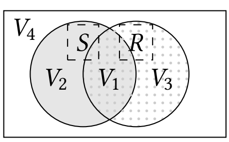


Fig. 1. Partition V

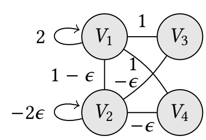


Fig. 2. CD

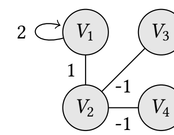


Fig. 3. ADS

Observation 1. *Given a graph $G(V,E)$, seed set R, for a set $S \subseteq V$, both key terms $\mathrm{vol}(S) - (1+\epsilon)\cdot \mathrm{vol}(S\setminus R)$ [9, 13–15] and $2|E(S)| - \mathrm{vol}(S\setminus R)$ [2] are simply weighted aggregations over edges on S and R. Specifically, partition set V to 4 sets based on their relations to S and R: $V_1 = S\cap R$, $V_2 = S\setminus R$, $V_3 = R\setminus S$ and $V_4 = \overline{R\cup S}$ (Figure 1). The edges on S and R can be accordingly partitioned into $4 + \binom{4}{2} = 10$ groups, e.g, one group could be $E_{2,2} = (V_2\times V_2)\cap E$ and another could be $E_{2,3} = (V_2 \times V_3)\cap E$. Assign each group of edges a weight, a real number in $\mathbb{R}$. The 10 real numbers form a configuration Ω. Figure 2 shows the non-zero weights of such a configuration Ω: for $e \in E_{2,2} \subseteq V_2\times V_2$, its weight, denoted as $w_{\Omega,R,S}(e)$, has $w_{\Omega,R,S}(e) = -2\epsilon$; for an edge $e \in E_{2,3} \subseteq V_2\times V_3$, its weight $w_{\Omega,R,S}(e) = -\epsilon$. It can be verified that $\mathrm{vol}(S) - (1+\epsilon)\cdot\mathrm{vol}(S\setminus R) = \mathrm{vol}(V_1) - \epsilon\mathrm{vol}(V_2) = 2|E_{1,1}| - 2\epsilon|E_{2,2}| + (1-\epsilon)|E_{1,2}| + |E_{1,3}| + |E_{1,4}| - \epsilon|E_{2,3}| - \epsilon|E_{2,4}| = \sum_{e\in E} w_{\Omega,R,S}(e)$. Similarly, when the configuration follows Figure 3, $2|E(S)| - \mathrm{vol}(S\setminus R) = \sum_{e\in E} w_{\Omega,R,S}(e)$. The key terms then have a uniform expression $\sum_{e\in E} w_{\Omega,R,S}(e)$.*

Configurations in Observation 1 encode local community search preferences in 10 real numbers and a uniform expression. We define the weight configuration below to formalize the encoding.

Definition 1 (Weight Configuration). *Let $\mathsf{I}_4 = \{1,2,3,4\}$. Let the **pair set** $\mathcal{I} = \{(i,j)|i,j \in \mathsf{I}_4\}$ be a set of 10 unordered pairs on I_4. A **weight configuration** Ω is a mapping from $\mathcal{I} \mapsto \mathbb{R}$, i.e., for any pair $p(i,j) \in \mathcal{I}$, $\Omega(p) \in \mathbb{R}$. When Ω is clear in the context, denote $\Omega(p(i,j))$ as ω_{ij} for simplicity.*

The weight configuration, consisting of 10 real numbers, is independent of the underlying graph G. Given a graph G and seed set R, to derive the objective function $f(S)$ for a set S, we conceptually partition the nodes and edges based on R and S, then apply the weight configuration for aggregation.

Definition 2 (Node and Edge Partitioning). *Given a graph $G(V,E)$, a seed set R and a set $S \subseteq V$, partition the nodes in V in 4 disjoint sets: $V_1 = S\cap R$, $V_2 = S\setminus R$, $V_3 = R\setminus S$ and $V_4 = \overline{R\cup S}$. Note that $S = V_1\cup V_2$, $R = V_1 \cup V_3$. Denote by $\mathcal{V}(S,R|G) = \{V_1,V_2,V_3,V_4\}$ the above partitioning of V. For each pair $p(i,j) \in \mathcal{I}$, define edge set $E_p = (V_i\times V_j)\cap E$. Denote the partitioning of E as $\mathcal{E}(S,R|G) = \{E_p|p\in\mathcal{I}\}$. When the graph G is clear in the context, denote $\mathcal{V}(S,R|G)$ as $\mathcal{V}(S,R)$ and $\mathcal{E}(S,R|G)$ as $\mathcal{E}(S,R)$. For simplicity, for $p(i,j) \in \mathcal{I}$, $E_{p(i,j)}$ is denoted as E_p or E_{ij} equivalently.*

With the conceptual edge partitioning above, define the key term under a weight configuration.

Definition 3 (Key Term under Weight Configuration). *Given graph $G(V, E)$ and weight configuration Ω, consider seed set R and an arbitrary set $S \subseteq V$. For an edge $e \in E$, as $\mathcal{E}(S, R|G)$ is a partitioning of E, there must be exactly one pair $p \in \mathcal{I}$ such that $e \in E_p$ where $E_p \in \mathcal{E}(S, R|G)$, we then define **the weight of** e as $w_{\Omega,R,S}(e|G) = \Omega(p)$. The **key term** $g_{\Omega,R}(S|G) = \sum_{e\in E} w_{\Omega,R,S}(e|G) = \sum_{p\in\mathcal{I},E_p\in\mathcal{E}(S,R|G)} \Omega(p)|E_p|$. When G is clear, denote $w_{\Omega,R,S}(e|G)$ as $w_{\Omega,R,S}(e)$ and $g_{\Omega,R}(S|G)$ as $g_{\Omega,R}(S)$.*

With the key term $g_{\Omega,R}(S)$ determined by the weight configuration Ω, the objective function of LCS plugs the key term in a community metric: the local density under Ω in G, denoted as $\rho_{\Omega,R}(S)$, has $\rho_{\Omega,R}(S) = \frac{g_{\Omega,R}(S)}{|S|}$ and the local conductance under Ω is $\Phi_{\Omega,R}(S) = \frac{\partial S}{g_{\Omega,R}(S)}$. *The weight configuration encodes the alignment between the seed set and a desirable community, thus allowing an interactive and exploratory* LCS *if an efficient computation is available.*

1.2 Strongly Local Computation

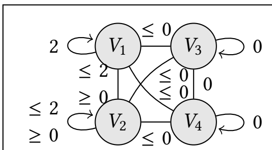
Any configuration can define an objective function of an LCS; however, it is not easy to design an efficient optimization algorithm. To better describe a "good" algorithm of an LCS, Orecchia and Zhu [9] used the word "local" and Veldt et al. [14] used the word "strongly local" to indicate a desirable property: an LCS algorithm is strongly local if its complexity is only dependent on the seed set and not the entire input graph. Usually, the complexity of existing strongly local algorithms is a function of the volume $\mathrm{vol}(R)$ of R. Finding a strongly local algorithm is not trivial.

Table 1 shows existing LCS with strongly local algorithms. Apart from one LCS [2] that is density-based, all others [9, 13–15] are conductance-based. Moreover, all existing strongly local algorithms are based on network flow under the same expansion framework (Algorithm 1). Specifically, let $\mathcal{A}$ be a network-flow-based algorithm reporting a local community in time polynomial to the graph size, called the global algorithm. The expansion framework starts with a core set $C_0 \doteq R$, iteratively expands the core set C_i and the corresponding *augment graph* [14] $L_i(V_i, E_i)$ until C_i can not be further expanded. The node set V_i contains nodes in C_i and their neighbors, i.e., $V_i = \mathcal{N}^+(C_i) \doteq \{u|(u,v) \in E, v \in C_i\} \cup C_i$; the edge set E_i includes all the edges with at least one node in C_i, i.e., $E_i = E^+(C_i) \doteq \{(u,v) \in E | v \in C_i\}$. Each iteration performs $\mathcal{A}$ on the augment graph L_i to get the LCS S_i which expands $C_{i+1} \doteq C_i \cup S_i$. It terminates when C_i stops expanding.

Algorithm 1: ExpansionFramework

Input: A graph $G = (V, E)$, seed node set $R \subseteq V$, weight configuration Ω, an algorithm $\mathcal{A}$ that reports a local community on a given graph.
Output: The local community S^* on G

```
i ← 0; S_0 ← ∅; C_0 ← R;
while true do
    V_i ← N⁺(C_i); E_i ← E⁺(C_i); L_i ← graph (V_i, E_i);
    S_i ← local community of R on L_i by calling 𝒜;
    if S_i ⊈ C_i then C_{i+1} ← C_i ∪ S_i; i ← i + 1 else
        break;
return S* ← S_i;
```



Constraint Graph

C1 $\omega_{11} = 2$,
C2 $\omega_{ij} = 0$ if $\min\{i, j\} > 2$,
C3 $\omega_{ij} \geq 0$ if $i, j \leq 2$,
C4 $\omega_{ij} \leq 0$ if $\max\{i, j\} > 2$,
C5 $\omega_{ij} \geq \omega_{i'j'}$ if $i \leq i'$ and $j \leq j'$.

Fig. 4. Density-based Weight Configurations C (Definition 4)

To prove that both the size of the eventual augment graph and the total number of iterations are dependent only on $\mathrm{vol}(R)$, properties of the flow network tailored specifically for the objective function are used, which we feel are hard to extend to other objective functions.

1.3 Density-based LCS and Our Results

Any weight configuration can define an objective function of LCS; however, not all configurations are aligned with the notion of density. Recall that the key term of the density $\rho(S)$ of a set $S \subseteq V$ is $|E(S)|$; we define density-based configurations $\mathcal{C}$ as follows. Note that since $S = V_1 \cup V_2$ in Definition 2, an edge in $E(S)$ has both end nodes in V_1 or V_2, its weight ω_{ij} should have $\max\{i, j\} \leq 2$; similarly, an edge in $E(\overline{S})$ should have its weight ω_{ij} with $\min\{i, j\} > 2$.

Definition 4 (Density-based Weight Configurations $\mathcal{C}$). *A weight configuration Ω is density-based if it satisfies conditions **C1-C5**. Denote by $\mathcal{C}$ the set of all density-based weight configurations. The constraint graph (Figure 4) reflects **C1-C4** and partially **C5**.*
C1 *$\omega_{11} = 2$ is set because i) scaling Ω does not change the result of* LCS*, and ii) the local density of S must reward the edges with both ends in $V_1 = R \cap S$. This setup is consistent to [2].*
C2 *is set because the density $\frac{|E(S)|}{|S|}$ focuses on the edges on $S = V_1 \cup V_2$, i.e., if an edges has both end nodes in $\overline{S}$, its weight should be* 0*.*
C3 *is set because the density should not penalize an edge in $E(S)$.*
C4 *is set because the density should not reward a cut edge in $S \times \overline{S}$.*
C5 *is set because the objective function on the edges e on S should reward more if e is more aligned to R. Specifically, nodes in $V_1 = S \cap R$ are preferred over those in $V_2 = S \setminus R$ and $V_3 = R \setminus S$, while the nodes in V_2, V_3 are preferred over those in $V_4 = \overline{S \cup R}$ as V_4 is not related to either S or R. For example, the weight ω_{11} of an edge in $E(V_1)$ should be no less than the weight ω_{12} of an edge between V_1 and V_2.*

For configurations $\Omega \in \mathcal{C}$ tailored to density-based LCS, recall the key term $g_{\Omega,R}(S)$ (Definition 3).

Problem 1 (Density-based LCS (DenLCS)). *Given a graph $G(V, E)$, a seed set R, a density-based weight configuration $\Omega \in \mathcal{C}$, for a set $S \subseteq V$, let $g_{\Omega,R}(S|G)$ be the key term of S under Ω and R defined in Definition 3, denote the local density of S under Ω and R as $\rho_{\Omega,R}(S|G) = \frac{g_{\Omega,R}(S|G)}{|S|}$. The maximum local density under Ω and R is $[\rho^*_{\Omega,R}|G] = \max_{S \subseteq V} \rho_{\Omega,R}(S|G)$. A subgraph $S^* \subseteq V$ is a local densest subgraph (*LDS*) under Ω and R on G if $\rho_{\Omega,R}(S^*|G) = [\rho^*_{\Omega,R}|G]$. When G is clear in the context, denote by $\rho_{\Omega,R}(S)$ the local density of S and by $\rho^*_{\Omega,R}$ the maximum local density.*

Example 1. *The traditional densest subgraph search is a special case of the general density-based* LCS*. Consider the configuration Ω with the following non-zero values $\omega_{11} = \omega_{12} = \omega_{22} = 2$. For any reference node set R, $\rho^*_{\Omega,R} = \max_{S \subseteq V} \rho_{\Omega,R}(S)$. Furthermore, anchored densest subgraph search [2] is a special density-based* LCS *under configuration Ω whose non-zero values are set in Figure 3.*

Similar to [2], we make an **assumption** on the input seed set R to ensure that $\rho^*_{\Omega,R}$ is not zero under all possible configurations, i.e., we assume that the seed set R has at least one edge $|E(R)| \geq 1$.

Definition 5 (Configuration Classification). *A configuration $\Omega \in \mathcal{C}$ is **global** if there is no possible strongly local solution for the density-based* LCS *under Ω; otherwise, Ω is **strongly local**.*

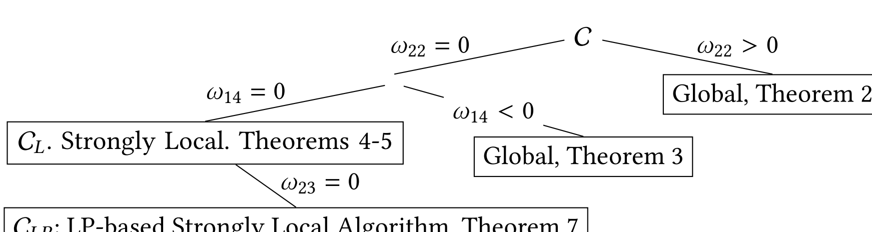


Fig. 5. The Classification of Configurations in $\mathcal{C}$ (Theorems 1&7)

The main challenges of density-based LCS are to identify what configurations are strongly local, and then to find a general strongly local solution for strongly local configurations. A practical solution for density-based LCS in real-world scenarios is highly desirable.

Our results. This paper classifies the configurations constructively: Figure 5 serves as a catalog of our main results. Theorem 1 is the general classification of configurations in C. Theorems 2-3 show that when $\omega_{22} > 0$ or $\omega_{14} < 0$, the configuration is global. Theorems 4-5 show that for all other configurations $C_L \subseteq C$, we have a general solution for density-based LCS that is correct and strongly local. They jointly prove Theorem 1. Theorem 7 shows that we have a Linear Programming (LP)-based strongly local solution for C_{LP}, the configurations in C_L whose $\omega_{23} = 0$. Theorems 1&7 constitute the main results of our paper. Section 1.4 applies our results to real-world scenarios.

1.4 Application

With our results shown in Figure 5, to get a density-based local community, one only needs to tune 3 out of 10 parameters of the configurations in C_L. In other words, because **C2** and **C5** ($\omega_{14} \le \omega_{13} \le 0$), all configurations in C_L have $\omega_{11} = 2$ and $\omega_{14} = \omega_{13} = \omega_{22} = \omega_{33} = \omega_{34} = \omega_{44} = 0$. Therefore, to use our LP-based strongly local algorithm on C_{LP}, only 2 parameters of the configurations are free for tuning, $\omega_{12} \in [0, 2]$ and $\omega_{24} \in (-\infty, 0]$. Denote by $\Omega_{x,y}$ a configuration in C_{LP} whose weights $\omega_{12} = x$ and $\omega_{24} = y$. We list below 4 interesting parameter settings that make sense. Denote, for any sets $X \subseteq Y \subseteq V$, by $\text{vol}_Y(X)$ the volume of X on the induced subgraph of Y.

- $\Omega_{1,0} = \frac{\text{vol}_S(S\cap R)}{|S|}$. The key term is the volume of $S \cap R$ on the induced subgraph of S.
- $\Omega_{2,0} = \frac{2|\{e\in E(S)|e\cap R\neq\emptyset\}|}{|S|}$. The key term counts the number of edges in $E(S)$ incident on R.
- $\Omega_{1,-1} = \frac{\text{vol}_S(S\cap R)-|E\cap(S\cap\overline{R})\times(\overline{S}\cap\overline{R})|}{|S|}$. The key term is the volume of $S \cap R$ on the induced subgraph of S, penalizing the number of cut edges of S in the induced subgraph of $\overline{R}$.
- $\Omega_{2,-1} = \frac{2|E(S)|-\text{vol}_{\overline{R}}(S\setminus R)}{|S|}$. The key term is $|2E(S)|$, penalizing each node v in $S \setminus R$ with its number of edges to $\overline{R}$.

The above 4 parameter settings shape different key terms and reasonably adapt the traditional density of S based on R. In practice, the rules of thumb in tuning parameters x and y are:

(1) Increase x to expand the search area — more nodes will be explored during the optimization;
(2) Decrease y to penalize more on high-degree nodes that are outside of R (consistent to ADS).

A case study in Appendix A.1 shows how an interactive search facilitates the above rules of thumb.

2 GLOBAL OR STRONGLY LOCAL?

Definition 6. *Let* $C_L = \{\Omega \in C|\Omega(2,2) = 0$ *and* $\Omega(1,4) = 0\}$ *be a subset of configurations in* C.

Theorem 1. *For any configuration* $\Omega \in C$, Ω *is strongly local **if and only if*** $\Omega \in C_L$.

The rest of this section serves as the proof of Theorem 1. Specifically, Theorems 2-3 in Section 2.1 prove that when the weight configuration $\Omega \in C$ has either $\omega_{2,2} > 0$ or $\omega_{1,4} < 0$, i.e., Ω is not in C_L, a local algorithms does not exist, and thus Ω is global. Theorems 4-5 in Section 2.2 jointly prove that there is a strongly local algorithm for density-based LCS under any configuration $\Omega \in C_L$.

For the simplicity of the discussion, we first define notations. Given a graph $G(V, E)$, for a node v, denote by $\deg(v)$ the degree of v in G. For a subset S of nodes in V, denote by $\text{vol}(S) = \sum_{v\in S} \deg(v)$ the volume of S. The set of neighbors of S including S is $\mathcal{N}^+(S) = \{u|\forall v \in S, \forall(v,u) \in E\} \cup S$. The net neighbor of S is $\mathcal{N}^-(S) = \mathcal{N}^+(S) \setminus S$. The set of incident edges of S is $E^+(S) = (S \times V) \cap E$. For two graphs $G_1(V_1, E_1)$ and $G_2(V_2, E_2)$, we say G_1 is a subgraph of G_2, denoted as $G_1 \subseteq G_2$, if $V_1 \subseteq V_2$ and $E_1 \subseteq E_2$. The following discussions consider graph $G(V, E)$ and seed set R.

2.1 Global Configurations

We first introduce two equivalences that shall be frequently used in the following discussions.

Lemma 1. *For $\forall a, b, c, d \in \mathbb{R}^+$ with $a \geq c > 0$, $b > d > 0$, for $\forall k \in \{1, -1\}$, Equivalence (1) holds; For $\forall a, b, c, d \in \mathbb{R}^+$ with $a, b, c, d > 0$, $\forall k \in \{1, -1\}$, Equivalence (2) holds. Moreover, Equivalences (1)-(2) hold when substituting $<$ with $\leq$ and $>$ with $\geq$ simultaneously. The proofs are in Appendix A.2.*

$$k\frac{a}{b} > k\frac{c}{d} \iff k\frac{a}{b} < k\frac{a-c}{b-d}. \quad (1) \qquad\qquad k\frac{a}{b} > k\frac{c}{d} \iff k\frac{a}{b} > k\frac{a+c}{b+d}. \quad (2)$$

Lemma 2. *For configuration $\Omega \in C$, if it has $\omega_{22} = 0$, then for any non-empty set $S \subseteq V$ and for any edge $e \in E \setminus E^+(R)$, $w_{\Omega,R,S}(e) \leq 0$. Please find the proofs in Appendix A.3.*

Theorem 2. *A configuration $\Omega \in C$ with $\omega_{22} > 0$ is global, i.e., there does not exist a strongly local algorithm for density-based* LCS*. Specifically, for any integer η such that $\eta > \frac{2}{\omega_{22}}$, there is a graph $G(V, E)$ and a seed set $R \subseteq V$ with* vol$(R) = 2$ *such that any subgraph* LDS *reports has $> \eta$ nodes.*

Proof. Consider the graph G in Figure 6 consisting of R, 2 nodes connected by an edge, and a k-clique where k can be any integer $> \eta$. Then, $|V| = 2 + k$. Denote by $V' = V \setminus R$ the nodes of the k-clique and S^* the LDS. We show $S^* = V'$ below in two steps: i) $S^* \cap R = \emptyset$ and ii) $S^* = V'$.

We prove $S^* \cap R = \emptyset$ by contradiction. Suppose $S^* \cap R \neq \emptyset$. The local density of V' $\rho_{\Omega,R}(V') = \frac{\omega_{22}k(k-1)}{2k} = \omega_{22}\frac{k-1}{2} \geq \omega_{22}\frac{\eta}{2} > \omega_{22} \times \frac{1}{\omega_{22}} = 1$, so $\rho_{\Omega,R}(S^*) \geq \rho_{\Omega,R}(V') > 1$. Further since R is a subgraph with only one edge, $\rho_{\Omega,R}(S^* \cap R) \leq \rho_{\Omega,R}(R) = 1$, which means $S^* \cap V' \neq \emptyset$ since if otherwise $\rho_{\Omega,R}(S^*) = \rho_{\Omega,R}(S^* \cap R) = 1 < \rho_{\Omega,R}(V')$. Also, as R has no edges to V' and $R \uplus V' = V$, $g_{\Omega,R}(S^*) = g_{\Omega,R}(S^* \cap V') + g_{\Omega,R}(S^* \cap R)$ and $|S^*| = |S^* \cap V'| + |S^* \cap R|$. Let $a = g_{\Omega,R}(S^*), b = |S^*|, c = g_{\Omega,R}(S^* \cap R), d = |S^* \cap R|, k = 1$, we can apply Equivalence (1) in Lemma 1 since $a \geq c > 0, b > d > 0, k\frac{a}{b} > 1 \geq k\frac{c}{d}$. Thus, $k\frac{a}{b} < k\frac{a-c}{b-d}$, i.e., $\frac{a}{b} < \frac{a-c}{b-d}$, that is, $\rho_{\Omega,R}(S^*) = \frac{g_{\Omega,R}(S^*)}{|S^*|} < \frac{g_{\Omega,R}(S^*)-g_{\Omega,R}(S^* \cap R)}{|S^*|-|S^* \cap R|} = \frac{g_{\Omega,R}(S^* \cap V')}{|S^* \cap V'|} = \rho_{\Omega,R}(S^* \cap V')$, contradicting that $\rho_{\Omega,R}(S^*) = \rho^*_{\Omega,R}$. Thus $S^* \cap R = \emptyset$.

We then show $S^* = V'$ by contradiction. Suppose $S^* \neq V'$, since $S^* \cap R = \emptyset$, we have $S^* \subseteq V'$ and $l \doteq |S^*| < k$. Recall that $\omega_{24} \leq 0$ (Definition 4 **C4**), then $\rho_{\Omega,R}(S^*) = \frac{\omega_{22}(l(l-1)/2)+\omega_{24}l(k-l)}{l} \leq \frac{\omega_{22}(l(l-1)/2)}{l} = \omega_{22}\frac{l-1}{2} < \omega_{22}\frac{k-1}{2} = \rho_{\Omega,R}(V') \leq \rho_{\Omega,R}(S^*)$, contradiction. Thus $S^* = V'$. □

According to Definition 4, $\omega_{22} \geq 0$ and $\omega_{14} \leq 0$; we next show that when $\omega_{22} = 0$, if $\omega_{14} < 0$, the configuration is also global.

Theorem 3. *Under a configuration $\Omega \in C$ with $\omega_{22} = 0$ and $\omega_{14} < 0$, there does not exist a strongly local algorithm for density-based* LCS*, i.e., Ω is global. Specifically, for any integer $\eta > 0$, there is a graph $G(V, E)$ with $|V| \geq \eta$ and a seed set $R \subseteq V$ with* vol$(R) = 2$ *such that* LDS *only can report G.*

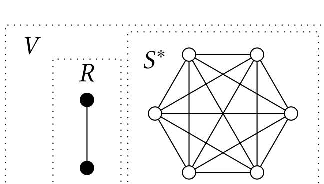


Fig. 6. Ω with $\omega_{22} > 0$ is Global

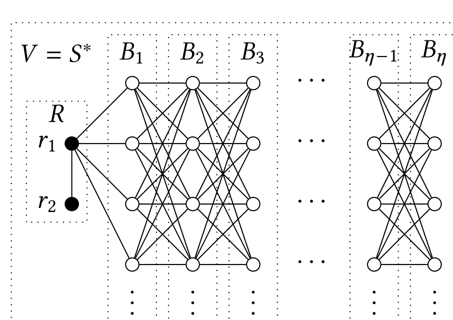


Fig. 7. Ω with $\omega_{22} = 0$ and $\omega_{14} < 0$ is Global

Proof. Consider the graph $G(V,E)$ in Figure 7 consisting of $R = \{r_1, r_2\}$ and η-layers of nodes: the i-th layer, $i \in [1,\eta]$, is a set B_i of m nodes where m is an arbitrary integer $m > \max\{2, -\frac{3}{\omega_{14}}\}$. r_1, r_2 have an edge; r_1 has an edge to each node in B_1; each node in B_i has an edge with each node in B_{i+1} for each $i \in [1, \eta-1]$. Note that according to **C5**, Definition 4, $\omega_{24} \le \omega_{14}$, so $m > \frac{-3}{\omega_{14}} \ge \frac{-3}{\omega_{24}}$. Define $\mathcal{S}^* = \{S \subseteq V | \rho_{\Omega,R}(S) = \rho^*_{\Omega,R}\}$ as the set of all the LDS s. We prove $\mathcal{S}^* = \{V\}$ in Lemmas 3-7. Lemma 3 shows that any local densest subgraph (S^*) of Figure 7 has a positive density; Lemma 4 shows that S^* includes R; Lemma 5 shows that for any S^*, there exists another subgraph that (not strictly) improves its local density by containing either all/none of nodes in a layer, for each layer; Lemma 6 shows that there is no S^* other than V with nodes containing all/none nodes in each layer; Lemma 7 shows that S^* is not the local densest subgraph if S^* contains part of nodes in one layer. Combining Lemma 6 and Lemma 7, we have $\mathcal{S}^* = \{V\}$. □

Lemma 3. $\rho^*_{\Omega,R} > 0$.

Proof. $\rho^*_{\Omega,R} \ge \rho_{\Omega,R}(V) = \frac{\omega_{11}+\omega_{12}m+\omega_{22}m^2(\eta-1)}{2+m\eta} \ge \frac{2}{2+m\eta} > 0$. □

Lemma 4. *For* $\forall S \in \mathcal{S}^*$, $r_1, r_2 \in S$.

Proof. We first show that $r_1 \in S$ for $\forall S \in \mathcal{S}^*$. If $r_1 \notin S$, the only possible edges having non-negative weights are in $E(S) = E(V_1 \cup V_2)$ where $V_1 \doteq S \cap R \subseteq \{r_2\}$ and $V_2 \doteq S \setminus R$, so $g_{\Omega,R}(S) \le \omega_{22}m^2(\eta-1) = 0$ and then $\rho_{\Omega,R}(S) \le 0$ contradicting Lemma 3 that $\rho_{\Omega,R}(S) = \rho^*_{\Omega,R} > 0$.

We then show that $r_2 \in S$ for $\forall S \in \mathcal{S}^*$ with $r_1 \in S$. Consider any $S \in \mathcal{S}^*$ with $r_1 \in S$ and $r_2 \notin S$. Denote $|S \cap B_1|$ as $k_1 \in [0,m]$. Thus, $|S| \ge 1 + k_1$. Consider the node partitioning $\{V_1, V_2, V_3, V_4\}$ and edge partitioning $\{E_p | p \in \mathcal{I}\}$ defined in Definition 2. $R = V_1 \cup V_3$ where $V_1 = R \cap S = \{r_2\}$ and $E^+(R) = (V_1 \cup V_3) \times V$. Following Definition 4, Ω has $\omega_{13} \le 0, \omega_{12} \le 2, \omega_{33} \le 0, \omega_{34} \le 0$. From Lemma 2, all edges in $E \setminus E^+(R)$ have weights no more than 0. Thus, $g_{\Omega,R}(S) \le \omega_{12}k_1 + \omega_{14}(m-k_1)$ as $E_{1,1} = \emptyset$. So as $\omega_{14} < 0$ (condition of Theorem 3), $g_{\Omega,R}(S) \le 2k_1$ and thus $\rho^*_{\Omega,R} = \rho_{\Omega,R}(S) \le \frac{g_{\Omega,R}(S)}{|S|} \le \frac{2k_1}{k_1+1} < 2$. Let $S' = S \cup \{r_2\}$. As adding r_2 to S only moves edge (r_1, r_2) from $E_{1,3}$ to $E_{1,1}$, $g_{\Omega,R}(S') = g_{\Omega,R}(S) + \omega_{11} - \omega_{13}$. As $\omega_{11} = 2$ and $\omega_{13} \le 0$, $\omega_{11} - \omega_{13} \ge 2 > \rho_{\Omega,R}(S) = \frac{g(S)}{|S|}$. Let $a = g_{\Omega,R}(S), b = |S|, c = \omega_{11} - \omega_{13}, d = 1, k = -1$. Then $k\frac{a}{b} > k\frac{c}{d}$. Since $a, b, c, d > 0$, by Lemma 1, Equivalence (2), we have $k\frac{a}{b} > k\frac{a+c}{b+d}$, thus $\frac{a}{b} < \frac{a+c}{b+d}$, that is, $\rho_{\Omega,R}(S) = \frac{g_{\Omega,R}(S)}{|S|} < \frac{g_{\Omega,R}(S)+\omega_{11}-\omega_{13}}{|S|+1} = \frac{g_{\Omega,R}(S')}{|S'|} = \rho_{\Omega,R}(S')$, contradicting $\rho_{\Omega,R}(S) = \rho^*_{\Omega,R}$. Thus, $r_2 \in S^*$. □

Lemma 5. *Consider any set $S \subseteq V$ of nodes in G with $r_1, r_2 \in S$. For any layer $B_i \subseteq V$, $i \in [1,\eta]$, $\rho_{\Omega,R}(S)$ can be increased (not strictly) by including B_i in S in an all-or-nothing manner, that is, $\rho_{\Omega,R}(S) \le \max\{\rho_{\Omega,R}(S \cup B_i), \rho_{\Omega,R}(S \setminus B_i)\}$.*

Proof. Denote by $k_i = |S \cap B_i|$ the number nodes S has in the i-th layer, for each int $i \in [1,\eta]$. Express with k_i, $i \in [1,\eta]$, key term $g_{\Omega,R}(S) = \omega_{11} + \omega_{12}k_1 + \omega_{14}(m-k_1) + \omega_{24}\sum_{i=1}^{\eta-1}(k_i(m-k_{i+1}) + k_{i+1}(m-k_i))$. Besides, $|S| = 2 + \sum_{i=1}^{\eta} k_i$. Fix an integer $i \in [1,\eta]$. $g_{\Omega,R}(S)$ can be rewritten as an expression $g_{\Omega,R}(S) = M_i k_i + C_i$ of variable k_i while treating other k_j, $j \ne i$, as constants. Here M_i is a function of k_j, $j \ne i$ and $C_i = g_{\Omega,R}(S) - M_i k_i$ is also a function of k_j, $j \ne i$. Let $C'_i = |S| - k_i = 2 + \sum_{j \in [1,\eta], j \ne i} k_j > 0$. $\rho_{\Omega,R}(S) = \frac{M_i k_i + C_i}{k_i + C'_i} = M_i + \frac{C_i - M_i C'_i}{k_i + C'_i}$, monotonic to variable k_i! In other words, when k_j is fixed for all the $j \in [1,\eta], j \ne i$, the sign of $C_i - M_i C'_i$ is then irrelevant to k_i and if $C_i - M_i C'_i > 0$ decreasing k_i to 0 gets a higher $\rho_{\Omega,R}(S)$; if $C_i - M_i C'_i < 0$, increasing k_i to m gets a higher $\rho_{\Omega,R}(S)$. Thus, $\rho_{\Omega,R}(S) \le \max\{\rho_{\Omega,R}(S \cup B_i), \rho_{\Omega,R}(S \setminus B_i)\}$. □

Lemma 6. *Denote by $\mathcal{S}^{01}$ the set of subgraphs in $\mathcal{S}^*$ with all-or-nothing intersections with each layer, i.e., $\forall S \in \mathcal{S}^{01}, \forall i \in [1,\eta], |B_i \cap S| \in \{0, m\}$. Then $\mathcal{S}^{01} = \{V\}$.*

Proof. We first prove that $\mathcal{S}^{01} \neq \emptyset$. Due to Lemma 3, $\mathcal{S}^* \neq \emptyset$. For any $S_0 \in \mathcal{S}^*$. Let $S = S_0$, then traverse all the layers: for each layer i, improve (not strictly) $\rho_{\Omega,R}(S)$ by either including to S or excluding from S all the nodes in B_i based on Lemma 5. The resulting S has $\rho_{\Omega,R}(S) \geq \rho_{\Omega,R}(S_0) = \rho^*_{\Omega,R}$ and thus $S \in \mathcal{S}^{01}$. We then prove that for $\forall S \in \mathcal{S}^{01}$, $B_1 \subseteq S$. Let S be a subgraph in $\mathcal{S}^{01}$. Thus, either $B_1 \subseteq S$ or $B_1 \cap S = \emptyset$. Lemma 4 means $r_1, r_2 \in S$ and $|S| \geq 2$. If $B_1 \cap S = \emptyset$, according to Lemma 2, we have key term $g_{\Omega,R}(S) \leq 2 - \omega_{14}m$. Since $m > -\frac{3}{\omega_{14}}$, $g_{\Omega,R}(S) < 0$, so $\rho_{\Omega,R}(S) < 0$. Due to Lemma 3, $\rho_{\Omega,R}(S) < \rho^*_{\Omega,R}$, contradiction. Therefore, $B_1 \subseteq S$. Finally, we prove that $\mathcal{S}^{01} = \{V\}$. Consider $S \in \mathcal{S}^{01}$. Based on the above discussion, $B_1 \subseteq S$. If there is an integer $h \in [2, \eta]$ such that a) $B_i \subseteq S$ for $\forall i < h$ and $B_h \cap S = \emptyset$. According to Definition 4 **C5** and the condition $\omega_{14} < 0$ in Theorem 3, $\omega_{24} \leq \omega_{14} < 0, \omega_{12} \leq 2$ and $m > \max\{2, -\frac{3}{\omega_{14}}\}$, thus, $g_{\Omega,R}(S) \leq 2 + \omega_{12}m + \omega_{24}m^2 \leq 2 + 2m + \omega_{24}m^2$. Note that $\omega_{24} < 0$ is a multiplier of m^2, so turning m to $-\frac{3}{\omega_{14}} < m$ will lead to $2 + 2m + \omega_{24}m^2 < 2 + 2m + \omega_{24}(-\frac{3}{\omega_{14}})m \leq 2 + 2m - 3m = 2 - m < 0$, contradicting Lemma 3 that $\frac{g_{\Omega,R}(S)}{|S|} = \rho^*_{\Omega,R} > 1$. □

Lemma 7. $\mathcal{S}^* \subseteq \mathcal{S}^{01}$.

Proof. Prove by contradiction. Assume that there is $S_0 \in \mathcal{S}^*$ such that $S_0 \notin \mathcal{S}^{01}$. We apply the same process in the proof of Lemma 6, i.e., traverse each layer $i \in [1, \eta]$, including B_i to / excluding B_i from S_{i-1} in an all-or-nothing manner to generate S_i and ensure that the non decreasing $\rho_{\Omega,R}(S_i)$. Eventually, $S_\eta \in \mathcal{S}^{01}$. According to Lemma 6, $\mathcal{S}^{01} = \{V\}$. Therefore, for every layer, we have $B_i \subseteq S_i$ as otherwise $S_\eta \neq V$. In other words, for each $i \in [1, \eta]$, $\{r1, r_2\} \cup B_1 \cup B_2 \cup \cdots \cup B_i \subseteq S_i$. Let h be the largest integer in $[1, \eta]$ such that $B_h \nsubseteq S_0$, i.e., not all the nodes in B_h are in S_0. In other words, $B_j \subseteq S_0$ for each $j > h$, and thus $B_j \subseteq S_{h-1}$ for each $j > h$. We thus have $V \setminus B_h \subseteq S_{h-1} \subsetneq S_h = V$. Now we calculate $\rho_{\Omega,R}(S_{h-1})$ which should be $\rho^*_{\Omega,R}$ since $S_0 \in \mathcal{S}^*$. Let $\delta = m - |B_h \cap S_0|$. S_{h-1} includes all nodes in V except δ nodes in B_h. According to Lemma 6, $h > 1$. Besides, according to the condition of Theorem 3, $\omega_{22} = 0$. Thus,

$$g_{\Omega,R}(S_{h-1}) \leq 2 + \omega_{12}m + \omega_{24}\delta m \leq g_{\Omega,R}(V) + \omega_{24}\delta m. \tag{3}$$

We then have $\rho_{\Omega,R}(S_{h-1}) = \frac{g_{\Omega,R}(S_{h-1})}{|V|-\delta} \leq \frac{g_{\Omega,R}(V)+\omega_{24}\delta m}{|V|-\delta}$. Note that $\frac{-\omega_{24}\delta m}{\delta} = -\omega_{24}m > 3 > \rho_{\Omega,R}(V) = \rho_{\Omega,R}(S^*)$. Let $a = g_{\Omega,R}(S_{h-1}), b = |S_{h-1}|, c = -\omega_{24}\delta m, d = \delta, k = -1$. Then $k\frac{a}{b} > k\frac{c}{d}$. Since $a, b, c, d > 0$, by Lemma 1, Equivalence (2), we then have $k\frac{a}{b} > k\frac{a+c}{b+d}$, thus $\frac{a}{b} < \frac{a+c}{b+d}$, that is, $\rho^*_{\Omega,R} = \rho_{\Omega,R}(S_{h-1}) = \frac{g_{\Omega,R}(S_{h-1})}{|S_{h-1}|} < \frac{g_{\Omega,R}(S_{h-1})-\omega_{24}\delta m}{|V|} \leq \frac{g_{\Omega,R}(V)}{|V|} = \rho_{\Omega,R}(V) = \rho^*_{\Omega,R}$ (the latter inequality comes from Equation 3), contradiction. □

2.2 Local Density Configurations

Recall that we have an ExpansionFramework defined in Algorithm 1. When it comes to the context of density-based LCS, we first define the input tuple (G, R, Ω) and base algorithm $\mathcal{A}$ as the input of ExpansionFramework, and then show that for all the configurations $\Omega \in C_L$, ExpansionFramework is both correct (Theorem 4) and strongly local (Theorem 5).

Definition 7 (Input for ExpansionFramework). *An **input tuple** (G, R, Ω) consists of a graph $G(V, E)$, a seed set $R \subseteq V$, a configuration Ω such that i) R has $|E(R)| > 0$ following the assumption made in Section 1.3, and ii) $\Omega \in C_L$ following Definition 6. A **base algorithm** $\mathcal{A}$ is a density-based* LCS *algorithm that returns, for an input tuple (G, R, Ω), an* LDS *S^* that is **maximal**, i.e., $\rho_{\Omega,R}(S^*) = \rho^*_{\Omega,R}$, and there is no subgraph S' with $\rho_{\Omega,R}(S') = \rho^*_{\Omega,R}$ and $S^* \subsetneq S'$.*

Lemma 8. *For an input tuple $(G(V, E), R, \Omega)$, a base algorithm $\mathcal{A}$ always exists. Specifically, denote by $|G| \doteq |E|$ the size of G. There is an algorithm $\mathcal{A}$ that reports the maximal* LDS *in $f^T_{\mathcal{A}}(|G|) = O(|G|2^{|G|})$ time and $f^S_{\mathcal{A}}(|G|) = O(|G|)$ space.*

Proof. Let v be an isolated node in G, i.e., v does not have an edge in E. Firstly, let S^* be an LDS, we prove $v \notin S^*$ by contradiction. Let $e(u,v) \in E$ be an edge with $u, v \in R$, e exists (Definition 7). Thus, $\rho^*_{\Omega,R} \geq \rho_{\Omega,R}(\{u,v\}) = 1$. If $v \in S^*$, then $0 < g_{\Omega,R}(S^*) = g_{\Omega,R}(S^* \setminus \{v\})$ and thus $\rho_{\Omega,R}(S^*) < \frac{g_{\Omega,R}(S^*)}{|S^*|-1} = \rho_{\Omega,R}(S^* \setminus \{v\})$. Thus, S^* is not an LDS of R on G, contradiction. Secondly, get V', the nodes in E, i.e., the non-isolated nodes in V in $O(|E|)$ time because $|V'| \leq 2|E|$. Enumerate all the $2^{|V'|}$ subsets of V, calculate $\rho_{\Omega,R}(S)$ for each $S \subseteq V'$ in $O(|E|)$ time, then report S^* that maximizes $\rho_{\Omega,R}(S)$ in $O(2^{|V'|}|E|) = O(2^{2|E|}|E|)$ time. Break ties by reporting the largest subgraph. □

Theorem 4. *For an input tuple (G, R, Ω) and base algorithm $\mathcal{A}$,* ExpansionFramework *is a correct algorithm for density-based* LCS*, i.e., the reported S^* of* ExpansionFramework *has $\rho_{\Omega,R}(S^*) = \rho^*_{\Omega,R}$.*

The proof of Theorem 4 is provided in Section 2.2.1.

Theorem 5. *For an input tuple (G, R, Ω) and base algorithm $\mathcal{A}$ defined in Lemma 8,* ExpansionFramework *is strongly local. In particular, the time and space complexities of* ExpansionFramework *are bounded by $O(\mathrm{vol}^2(R) f^T_{\mathcal{A}}(\mathrm{vol}^4(R)))$ and $O(f^S_{\mathcal{A}}(\mathrm{vol}^4(R)))$ respectively if the weights in Ω are integers, and by $O(f^T_{\mathcal{A}}(\mathrm{vol}(R)))$ and $O(f^S_{\mathcal{A}}(\mathrm{vol}(R)))$ respectively if Ω has $\omega_{24} = 0$.*

Section 2.2.2 proves Theorem 5. Specifically, Lemma 17 covers the input tuples where Ω has $\omega_{24} = 0$, and Lemma 23 covers the input tuples where Ω has $\omega_{24} < 0$.

2.2.1 The proof of Theorem 4. Lemma 9 shows that under the same weight configuration, if one subgraph of G is a subgraph of another subgraph of G, then for each edge they share, the weights of the edge are the same for the two subgraphs. Lemma 10 shows that when ExpansionFramework terminates, the LDS on the last working graph has the same local density as the LDS of G. Lemma 11 shows that for any set S of nodes, the local density of S on the working graph will not increase along expanding working graphs in the iterations. We then combine Lemma 10 and Lemma 11 to prove Theorem 4. Note that the lemmas in this section will be used in Section 2.2.2.

Lemma 9. *Consider two input tuples $(G''(V'', E''), R, \Omega)$ and $(G'(V', E'), R, \Omega)$ sharing the same set of seed nodes R and configuration Ω such that G' is a subgraph of G''. We abuse R to denote the induced subgraph of nodes R on G'. Thus, we have $R \subseteq G' \subseteq G''$. For a non-empty set $S \subseteq V'$ and any edge $e \in E'$, the weight of edge e on G' for R and S and the weight of e on G'' under R and S are the same, i.e., $w_{\Omega,R,S}(e|G') = w_{\Omega,R,S}(e|G'')$.*

Proof. By Definition 2, denote by $\{V'_1, V'_2, V'_3, V'_4\}$ the partition of $\mathcal{V}(S, R|G')$ of nodes in G' and $\{V''_1, V''_2, V''_3, V''_4\}$ the partition of $\mathcal{V}(S, R|G'')$ of nodes in G''. By definition, $V'_1 = R \cap S = V''_1$, $V'_2 = S \setminus R = V''_2$, $V'_3 = R \setminus S = V''_3$, and $V'_4 = V' \setminus (R \cup S) \subseteq V'' \setminus (R \cup S) = V''_4$. Now denote the edge partitioning $\mathcal{E}(S, R|G')$ of G' as $\{E'_p | p \in \mathcal{I}\}$ and $\mathcal{E}(S, R|G'')$ of G'' as $\{E''_p | p \in \mathcal{I}\}$. For each $p(i,j) \in \mathcal{I}$ and any edge $e(u,v) \in E'_p$ with $u \in V'_i$ and $u \in V'_j$, we thus have $u \in V''_i$, $v \in V''_j$ and $e(u,v) \in E''$. Thus $e \in E''_p$ and $w_{\Omega,R,S}(e|G') = w_{\Omega,R,S}(e|G'') = \Omega(p)$. Since $\{E'_p | p \in \mathcal{I}\}$ is a partitioning of E', for each $e \in E'$, $w_{\Omega,R,S}(e|G') = w_{\Omega,R,S}(e|G'')$. □

Lemma 10. *Consider input tuple $(G(V, E), R, \Omega)$ with $\mathcal{A}$ defined in Lemma 8. Apply* ExpansionFramework *and denote by S^* the output subgraph and $L_k(V_k, E_k)$ the working graph of the last iteration. Then the local density of S^* on L_k is the same as the local density of S^* on G, i.e., $\rho_{\Omega,R}(S^*|L_k) = \rho_{\Omega,R}(S^*|G)$.*

Proof. By definition, $\rho_{\Omega,R}(S^*|G) = \frac{g_{\Omega,R}(S^*|G)}{|S^*|}$ and $\rho_{\Omega,R}(S^*|L_k) = \frac{g_{\Omega,R}(S^*|L_k)}{|S^*|}$ where $g_{\Omega,R}(S^*|G) = \sum_{e \in E} w_{\Omega,R,S^*}(e|G)$ and $g_{\Omega,R}(S^*|L_k) = \sum_{e \in E_k} w_{\Omega,R,S^*}(e|L_k)$. Apply Lemma 9 with G' being L_k and G'' being G, $w_{\Omega,R,S^*}(e|G) = w_{\Omega,R,S^*}(e|L_k)$ for each $e \in E_k$. Besides, when ExpansionFramework terminates, $E^+(S^*) \subseteq E_k$. By Constraint **C2**, Definition 4, $w_{\Omega,R,S^*}(e|G) = 0$ for $\forall e \in E \setminus E^+(S^*) \supseteq$

$E \setminus E_k$. Thus $g_{\Omega,R}(S^*|G) = \sum_{e\in E_k} w_{\Omega,R,S^*}(e|G) + \sum_{e\in E\setminus E_k} w_{\Omega,R,S^*}(e|G) = \sum_{e\in E_k} w_{\Omega,R,S^*}(e|L_k) + 0 = g_{\Omega,R}(S^*|L_k)$, and therefore, $\rho_{\Omega,R}(S^*|G) = \rho_{\Omega,R}(S^*|L_k)$. □

Lemma 11. *Consider an input tuple $(G(V,E), R, \Omega)$ and a set $S \subseteq V$. Denote by $L_0 = (\mathcal{N}^+(R), E^+(R))$ a subgraph of G. For any graph $G'(V', E')$ such that $L_0 \subseteq G' \subseteq G$, then the local density of S on G is no larger than the local density of $S \cap V'$ on G', i.e., $\rho_{\Omega,R}(S|G) \leq \rho_{\Omega,R}(S \cap V'|G')$. If $S \setminus V' \neq \emptyset$, then $\rho_{\Omega,R}(S|G) < \rho_{\Omega,R}(S \cap V'|G')$.*

Proof. Let S' be $S \cap V'$. By definition, $\rho_{\Omega,R}(S|G) = \frac{g_{\Omega,R}(S|G)}{|S|}$ and $\rho_{\Omega,R}(S'|G') = \frac{g_{\Omega,R}(S'|G')}{|S'|}$. Because $S' \subseteq S$, $|S| \geq |S'|$. By Lemma 9, all edges in E' have the same weight in both G and G', so $g_{\Omega,R}(S|G) - g_{\Omega,R}(S|G') = \sum_{e\in E\setminus E'} w_{\Omega,R,S}(e|G)$. By Lemma 2, $w_{\Omega,R,S}(e|G) \leq 0$ for all $e \in E \setminus E^+(R)$. Note that $L_0 \subseteq G'$, so $E^+(R) \subseteq E'$, thus $g_{\Omega,R}(S|G) - g_{\Omega,R}(S|G') = \sum_{e\in E\setminus E'} w_{\Omega,R,S}(e|G) \leq 0$. Thus, $\rho_{\Omega,R}(S|G) \leq \rho_{\Omega,R}(S'|G')$. If $S \setminus V' \neq \emptyset$, then $S' \subsetneq S$, $|S| > |S'|$ and thus $\rho_{\Omega,R}(S|G) < \rho_{\Omega,R}(S'|G')$. □

Denote by S^* the output of ExpansionFramework. Denote by $L_k(V_k, E_k)$ the working subgraph of the iteration before termination. We prove Theorem 4 by contradiction, assume there exists $S' \neq S^*$ s.t. $\rho_{\Omega,R}(S'|G) = [\rho^*_{\Omega,R}|G] > \rho_{\Omega,R}(S^*|G)$. Consider the local density $\rho_{\Omega,R}(S' \cap V_k|L_k) = \frac{g_{\Omega,R}(S'\cap V_k|L_k)}{|S'\cap V_k|}$. Apply Lemma 11 by letting G' be L_k, we have $\rho_{\Omega,R}(S' \cap V_k|L_k) \geq \rho_{\Omega,R}(S'|G)$. Also by applying Lemma 10, $\rho_{\Omega,R}(S^*|G) = \rho_{\Omega,R}(S^*|L_k)$. Combining the assumption $\rho_{\Omega,R}(S'|G) > \rho_{\Omega,R}(S^*|G)$, we have $\rho_{\Omega,R}(S' \cap V_k|L_k) > \rho_{\Omega,R}(S^*|L_k)$, contradicting the fact that S^* is the local denset subgraph on L_k, i.e., $\rho_{\Omega,R}(S^*|L_k) = [\rho^*_{\Omega,R}|L_k]$. This proves Theorem 4.

2.2.2 Proof of Theorem 5. In this section, Lemma 13 shows that the maximal density-based LCS is unique. Lemma 17 proves strong locality of configurations Ω with $\omega_{24} = 0$. For Ω with $\omega_{24} < 0$, Lemma 15, Lemma 18, Lemmas 21 & 22 show that for the maximal density-based LCS S of the last iteration, the size of S, the maximum degree of nodes in S, the number of iterations, and the size of the last working graph, are all bounded by polynomials of vol(R). Lemmas 17 and 23 combine the above bounds to prove the strong locality of ExpansionFramework in both space and time.

Definition 8. *Define set $\mathcal{I}_L = \{(1,1), (1,2), (2,3), (2,4)\}$ of pairs.*

Lemma 12. *Given input tuple $(G(V,E), R, \Omega)$, denote by $\{E_p | p \in \mathcal{I}\}$ the edge partitioning of $\mathcal{E}(S,R)$. For any non-empty set $S \subseteq V$, we have the key term $g_{\Omega,R}(S) = \sum_{p\in\mathcal{I}_L} \Omega(p)|E_p|$.*

Proof. Note that based on Definition 7, $\Omega \in \mathcal{C}_L$. $g_{\Omega,R}(S) = \sum_{p\in\mathcal{I}} \Omega(p)|E_p|$, so it suffices for us to show that for any $p \in \mathcal{I} \setminus \mathcal{I}_L$, $\Omega(p) = 0$. Since $\Omega \in \mathcal{C}_L$, (1) $\omega_{33}, \omega_{34}, \omega_{44}$ are 0 by Definition 4 **C2**; (2) ω_{22}, ω_{14} are 0 because of Definition 6; (3) $\omega_{13} = 0$ because $\omega_{13} \geq \omega_{14} = 0$ by Constraint **C5** and $\omega_{13} \leq 0$ by Constraint **C4**. Thus $g_{\Omega,R}(S) = \sum_{p\in\mathcal{I}} \Omega(p)|E_p| = \sum_{p\in\mathcal{I}_L} \Omega(p)|E_p|$. □

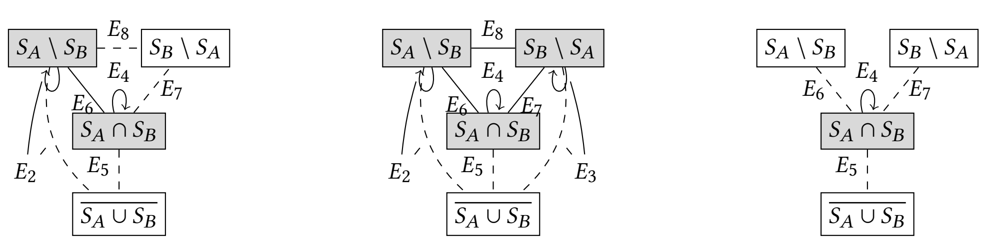
Lemma 13. *Consider an input tuple $(G(V,E), R, \Omega)$ and two* LDS *s $S_A, S_B \subseteq V(G)$ such that $\rho_{\Omega,R}(S_A) = \rho_{\Omega,R}(S_B) = \rho^*_{\Omega,R}$. We have $\rho_{\Omega,R}(S_A \cup S_B) = \rho^*_{\Omega,R}$.*

Proof. As $\rho^*_{\Omega,R}$ is optimal, $\rho_{\Omega,R}(S_A \cup S_B) \leq \rho^*_{\Omega,R}$. To show $\rho_{\Omega,R}(S_A \cup S_B) \geq \rho^*_{\Omega,R}$, we disjointly partition the edges E into 8 edge sets (it can be verified by plugging S with S_A and R with S_B in Figure 1 that the 8 sets below disjointly cover E):

- $E_1 \doteq E \cap (\overline{S_A \cup S_B} \times \overline{S_A \cup S_B})$,
- $E_2 \doteq E \cap ((S_A \setminus S_B) \times (V \setminus S_B))$,
- $E_3 \doteq E \cap ((S_B \setminus S_A) \times (V \setminus S_A))$,
- $E_4 \doteq E \cap ((S_A \cap S_B) \times (S_A \cap S_B))$,
- $E_5 \doteq E \cap ((S_A \cap S_B) \times \overline{S_A \cup S_B}))$,
- $E_6 \doteq E \cap ((S_A \cap S_B) \times (S_A \setminus S_B)))$,
- $E_7 \doteq E \cap ((S_A \cap S_B) \times (S_B \setminus S_A)))$,
- $E_8 \doteq E \cap ((S_A \setminus S_B) \times (S_B \setminus S_A))$.

For simplicity, denote $w_A(e) \doteq w_{\Omega,R,S_A}(e|G)$, $w_B(e) \doteq w_{\Omega,R,S_B}(e|G)$, $w_\cap(e) \doteq w_{\Omega,R,S_A\cap S_B}(e|G)$ and $w_\cup(e) \doteq w_{\Omega,R,S_A\cup S_B}(e|G)$. We first build up relations among edge weights in each case:

(1) Consider $e \in E_1$. By Constraint **C2**, Definition 4 (i.e., any edge with both ends outside S has weight 0), $w_A(e) = w_B(e) = w_\cup(e) = w_\cap(e) = 0$.
(2) Consider $e(u, v) \in E_2$. Thus, both u and v are not in S_B and therefore, $u \in S_A$ iff $u \in S_A \cup S_B$ and $v \in S_A$ iff $v \in S_A \cup S_B$, and $w_B(e) = w_\cap(e) = 0$. Denote the 2 partitions $\mathcal{E}(E, S_A, R, G) = \{E_p^A | p \in \mathcal{I}\}$ and $\mathcal{E}(E, S_A \cup S_B, R, G) = \{E_p^\cup | p \in \mathcal{I}\}$. We thus have for each $p \in \mathcal{I}$, $e \in E_p^A$ iff $e \in E_p^\cup$ and it follows that $w_A(e) = w_\cup(e)$.
(3) Consider $e \in E_3$: symmetric to case E_2. We have $w_B(e) = w_\cup(e)$ and $w_A(e) = w_\cap(e) = 0$ (Constraint **C2**, Definition 4).
(4) Consider $e(u, v) \in E_4$. Denote the 4 partitions $\mathcal{E}(E, S_A, R, G) = \{E_p^A | p \in \mathcal{I}\}$, $\mathcal{E}(E, S_B, R, G) = \{E_p^B | p \in \mathcal{I}\}$, $\mathcal{E}(E, S_A \cup S_B, R, G) = \{E_p^\cup | p \in \mathcal{I}\}$ and $\mathcal{E}(E, S_A \cap S_B, R, G) = \{E_p^\cap | p \in \mathcal{I}\}$. $e \in E_4$, i.e., both $u, v \in S_A \cap S_B$, thus for each $p \in \mathcal{I}$, $e \in E_p^A$ iff $e \in E_p^B$ iff $e \in E_p^\cup$ iff $e \in E_p^\cap$. and then $w_A(e) = w_B(e) = w_\cup(e) = w_\cap(e) \geq 0$ (by Constraint **C3**, Definition 4).
(5) Consider $e(u, v) \in E_5$. Consider 4 partitionings $\mathcal{E}(E, S_A, R, G) = \{E_p^A | p \in \mathcal{I}\}$, $\mathcal{E}(E, S_B, R, G) = \{E_p^B | p \in \mathcal{I}\}$, $\mathcal{E}(E, S_A \cup S_B, R, G) = \{E_p^\cup | p \in \mathcal{I}\}$ and $\mathcal{E}(E, S_A \cap S_B, R, G) = \{E_p^\cap | p \in \mathcal{I}\}$. Since $e(u, v) \in E_5$, w.l.o.g., assume that $u \in S_A \cap S_B$ and $v \in \overline{S_A \cup S_B}$, then for any $S \in \{S_A, S_B, S_A \cup S_B$ and $S_A \cap S_B\}$, we have $u \in S, v \notin S$. It follows that, for each $p \in \mathcal{I}$, $e \in E_p^A$ iff $e \in E_p^B$ iff $e \in E_p^\cup$ iff $e \in E_p^\cap$, therefore $w_A(e) = w_B(e) = w_\cup(e) = w_\cap(e) \leq 0$ (by Constraint **C4**).
(6) Consider $e \in E_6$. Denote the 2 partitions $\mathcal{E}(E, S_A, R, G) = \{E_p^A | p \in \mathcal{I}\}$ and $\mathcal{E}(E, S_A \cup S_B, R, G) = \{E_p^\cup | p \in \mathcal{I}\}$. $e \in E_4$, i.e., both $u, v \in S_A$, thus for each $p \in \mathcal{I}$, $e \in E_p^A$ iff $e \in E_p^\cup$ and it follows that $w_A(e) = w_\cup(e) \geq 0$ (Constraints **C4**). Besides, by Constraint **C4** (as e as one node inside S and one node outside S), $w_B(e) \leq 0$ and $w_\cap(e) \leq 0$.
(7) Consider $e \in E_7$. Symmetric to case E_6. $w_B(e) = w_\cup(e) \geq 0$, $w_A(e) \leq 0$ and $w_\cap(e) \leq 0$.
(8) Consider $e \in E_8$. By Constraint **C4**, $w_A(e) \leq 0$, $w_B(e) \leq 0$. By **C3**, $w_\cup(e) \geq 0$. By Constraint **C2**, $w_\cap(e) = 0$.



(a) $S = S_A$ (symmetric to the case of $S = S_B$)
(b) $S = S_A \cup S_B$
(c) $S = S_A \cap S_B$

Fig. 8. Edge type breakdown for S_A, S_B, $S_A \cup S_B$ and $S_A \cap S_B$.

Figure 8 gives an overview of edge weights for each edge type when the seed set S is S_A, S_B, $S_A \cup S_B$ and $S_A \cap S_B$. Solid edges have non-negative weights; dashed edges have non-positive weights. Now we introduce notations based on the above weight relations to simplify the density analysis. Let $W_2 \doteq \sum_{e \in E_2} w_A(e) = \sum_{e \in E_2} w_\cup(e)$. Denote by $W_3 \doteq \sum_{e \in E_3} w_B(e) = \sum_{e \in E_3} w_\cup(e)$ and $\sum_{e \in E_3} w_A(e) = \sum_{e \in E_3} w_\cap(e) = 0$. Denote $W_4^+ \doteq \sum_{e \in E_4} w_A(e) = \sum_{e \in E_4} w_B(e) = \sum_{e \in E_4} w_\cup(e) = \sum_{e \in E_4} w_\cap(e) \geq 0$. The + is adopted because $W_4^+ \geq 0$. Denote $W_5^- \doteq \sum_{e \in E_5} w_A(e) = \sum_{e \in E_5} w_B(e) = \sum_{e \in E_5} w_\cup(e) = \sum_{e \in E_5} w_\cap(e) \leq 0$. The – is adopted because $W_5^- \leq 0$. Denote $W_6^+ \doteq \sum_{e \in E_6} w_A(e) = \sum_{e \in E_6} w_\cup(e) \geq 0$. Denote $W_6^- \doteq \sum_{e \in E_6} w_B(e) = \sum_{e \in E_6} w_\cap(e) \leq 0$. Denote $W_7^+ \doteq \sum_{e \in E_7} w_B(e) = \sum_{e \in E_7} w_\cup(e) \geq 0$, $W_7^- \doteq \sum_{e \in E_6} w_A(e) = \sum_{e \in E_7} w_\cap(e) \leq 0$. Denote $W_8^A \doteq \sum_{e \in E_8} w_A(e) \leq 0$, $W_8^B = \sum_{e \in E_8} w_B(e) \leq 0$ and $W_8^+ = \sum_{e \in E_8} w_\cup(e) \geq 0$, and $\sum_{e \in E_8} w_\cap(e) = 0$. Thus, $-W_8^A - W_8^B + W_8^+ \geq 0$.

Now list the 4 key terms based on the weights of the 8 edge sets:

- $g_{\Omega,R}(S_A) = W_2 + W_4^+ + W_5^- + W_6^+ + W_7^- + W_8^A$,
- $g_{\Omega,R}(S_B) = W_3 + W_4^+ + W_5^- + W_6^- + W_7^+ + W_8^B$,
- $g_{\Omega,R}(S_A \cup S_B) = W_2 + W_3 + W_4^+ + W_5^- + W_6^+ + W_7^+ + W_8^+$,
- $g_{\Omega,R}(S_A \cap S_B) = W_4^+ + W_5^- + W_6^- + W_7^-$.

Thus, $\rho_{\Omega,R}(S_A \cup S_B) = \frac{g_{\Omega,R}(S_A \cup S_B)}{|S_A \cup S_B|} = \frac{g_{\Omega,R}(S_A \cup S_B)}{|S_A|+|S_B|-|S_A \cap S_B|} =$

$$\frac{g_{\Omega,R}(S_A) + g_{\Omega,R}(S_B) - W_4^+ - W_5^- - W_6^- - W_7^- - W_8^A - W_8^B + W_8^+}{|S_A| + |S_B| - |S_A \cap S_B|} \tag{4}$$

$$\geq \frac{g_{\Omega,R}(S_A) + g_{\Omega,R}(S_B) - W_4^+ - W_5^- - W_6^- - W_7^-}{|S_A| + |S_B| - |S_A \cap S_B|}. \tag{5}$$

Let $a = g_{\Omega,R}(S_A) + g_{\Omega,R}(S_B), b = |S_A| + |S_B|, c = W_4^+ + W_5^- + W_6^- + W_7^-, d = |S_A \cap S_B| \geq 0, k = 1$, consider the following cases:

(1) If $S_A \cup S_B = S_A$ or S_B, the lemma is correct as $\rho_{\Omega,R}(S_A \cup S_B) \geq \min\{\rho_{\Omega,R}(S_A), \rho_{\Omega,R}(S_B)\} = \rho^*_{\Omega,R}$;

(2) Else if $c \leq 0$, the lemma is correct as $\rho_{\Omega,R}(S_A \cup S_B) = \frac{g_{\Omega,R}(S_A)+g_{\Omega,R}(S_B)-c}{|S_A|+|S_B|-d} \geq \frac{g_{\Omega,R}(S_A)+g_{\Omega,R}(S_B)}{|S_A|+|S_B|} = \rho^*_{\Omega,R}$;

(3) Else, we have $S_A \setminus S_B \neq \emptyset$, $S_B \setminus S_A \neq \emptyset$, and $c > 0$, then we have
   (a) $d > 0$, because otherwise $S_A \cap S_B = \emptyset$, so no edge is incident on $S_A \cap S_B$ and by Constraint **C2**, Definition 4, $g_{\Omega,R}(S_A \cap S_B) = W_4^+ + W_5^- + W_6^- + W_7^- = d = 0$, contradicts $d > 0$;
   (b) $b > d$, because $b \geq |S_A \cup S_B| > |S_A| \geq d$;
   (c) $a > 0$, because by Lemma 14, $\rho^*_{\Omega,R} \geq 1$, thus $a \geq b > 0$;
   (d) $\frac{a}{b} \geq \frac{c}{d}$, because $\frac{a}{b} = \rho^*_{\Omega,R} \geq \rho_{\Omega,R}(S_A \cap S_B) = \frac{c}{d}$;
   (e) $a > c$, because $\frac{a}{b} \geq \frac{c}{d} \implies a \geq \frac{b}{d}c \implies a > c$ by applying the above (b) and (d).
   Now that we have $a \geq c > 0$, $b > d > 0$, $k\frac{a}{b} \geq k\frac{c}{d}$, according to Lemma 1 equivalence (1), we have $k\frac{a}{b} \geq k\frac{c}{d} \iff k\frac{a}{b} \leq k\frac{a-c}{b-d}$. As $k\frac{a}{b} \geq k\frac{c}{d}$, and by Eqn (5), we have $\rho_{\Omega,R}(S_A \cup S_B)$

$$\geq \frac{g_{\Omega,R}(S_A) + g_{\Omega,R}(S_B) - W_4^+ - W_5^- - W_6^- - W_7^-}{|S_A| + |S_B| - |S_A \cap S_B|} = k\frac{a-c}{b-d} \geq k\frac{a}{b} = \frac{g_{\Omega,R}(S_A) + g_{\Omega,R}(S_B)}{|S_A| + |S_B|} = \rho^*_{\Omega,R},$$

thus concludes the proof that $\rho_{\Omega,R}(S_A \cup S_B) = \rho^*_{\Omega,R}$. □

Lemma 13 shows that for $\forall \Omega \in C_L$, the maximal LDS is unique, which is also the maximum LDS.

Lemma 14. *Given an input tuple (G, R, Ω) and a base algorithm $\mathcal{A}$, apply* ExpansionFramework*; denote by k the max value of i before terminating the while-loop (Line 2-Line 5) and denote, for each $i \in [0, k]$, by L_i the working graph and by S_i the* LDS *of L_i (returned by $\mathcal{A}$). We then have $\rho_{\Omega,R}(S_i|L_i) \geq 1$, for each $i \in [0, k]$.*

Proof. By Definition 7, R is required to have an edge, i.e., $|E(R)| > 0$. Let $e(u, v)$ be an arbitrary edge in R. We show below that the set $S = \{u, v\}$ has local density 1 on every L_i. Consider L_i, $i \in [0, k]$. Denote by $\{E_p^i | p \in \mathcal{I}\}$ the edge partitioning of $\mathcal{E}(S, R|L_i)$ defined in Definition 2. The local density $\rho_{\Omega,R}(S|L_i) = \frac{g_{\Omega,R}(S|L_i)}{|S|}$ where $g_{\Omega,R}(S|L_i) = \sum_{p \in \mathcal{I}} \omega_p |E_p^i| = \sum_{p \in \mathcal{I}_L} \Omega(p)|E_p^i|$ by Lemma 12. Furthermore, as S is a subset of R, in the node partitioning of L_i, $V_2^i \doteq S \setminus R = \emptyset$. Thus, for $\forall p \in \{(1, 2), (2, 3), (2, 4)\} = \mathcal{I}_L \setminus \{(1, 1)\}$, $|E_p| \leq |V_2^i \times V| = 0$. Thus, $g_{\Omega,R}(S|L_i) = \omega_{11}|E_{11}^i| = 2$ as $\omega_{11} = 2$ for $\Omega \in C_L$ and $E_{11} = \{e(u, v)\}$. Thus $\rho_{\Omega,R}(S|L_i) = 1$. Since S_i is the LDS on L_i, $\rho_{\Omega,R}(S_i|L_i) \geq \rho_{\Omega,R}(S|L_i) = 1$. □

Lemma 15. *Consider an input tuple (G, R, Ω). Denote by S^* the* LDS *of the input tuple, then $|S^*| \leq g_{\Omega,R}(S^*) \leq 2\mathrm{vol}(R)$, i.e., $|S^*| = O(\mathrm{vol}(R))$. For the ease of the following discussions, let U_s be an alias of $2\mathrm{vol}(R)$, we then have $|S^*| = O(U_s)$.*

Proof. By Lemma 2, only edges $e \in E^+(R)$ can have positive weights and by Constraint **C1** and **C3**, Definition 4, $w_{\Omega,R,S}(e) \leq 2$. By Lemma 14, let S_k be the local densest graph on L_k, the last working graph of ExpansionFramework, $\rho_{\Omega,R}(S_k|L_k) \geq 1$; further based on Lemma 10, $\rho_{\Omega,R}(S_k|L_k) = \rho_{\Omega,R}(S_k|G)$, thus $\rho_{\Omega,R}(S_k|G) > 1$. Then, $\rho_{\Omega,R}(S^*) = \frac{g_{\Omega,R}(S^*)}{|S^*|} \geq \rho_{\Omega,R}(S_k|G) \geq 1$ and thus $|S^*| \leq g_{\Omega,R}(S^*) \leq \sum_{e\in E^+(R)} w_{\Omega,R,S}(e) \leq 2|E^+(R)| \leq 2\mathsf{vol}(R)$. □

Lemma 16. *Consider input tuple $(G(V,E),R,\Omega)$ with base algorithm $\mathcal{A}$ defined in Lemma 8. If Ω has $\omega_{24} = 0$, then for any subgraph $L'(V',E')$ of G that satisfies $L_0(\mathcal{N}^+(R),E^+(R)) \subseteq L'$, then any $S \subseteq \mathcal{N}^+(R)$ has $\rho_{\Omega,R}(S|L') = \rho_{\Omega,R}(S|L_0) = \rho_{\Omega,R}(S|G)$.*

Proof. Let $\{E'_p|p \in \mathcal{I}\}$ be the edge partitioning $\mathcal{E}(S,R|L')$, and $\{E^0_p|p \in \mathcal{I}\}$ be $\mathcal{E}(S,R|L_0)$. Note that in the node partitioning $\mathcal{V}(S,R|L')$ and $\mathcal{V}(S,R|L_0)$, they share the same $V_0 = S \cap R$, $V_1 = S \setminus R$. Furthermore, $V_2 = R \setminus S$ and $(V_1 \times V_1) \cap E, (V_1 \times V_2) \cap E, (V_2 \times V_3) \cap E \subseteq E^+(R) \subseteq E'$. Thus $E'_{11} = E^0_{11}$, $E'_{12} = E^0_{12}$, and $E'_{12} = E^0_{13}$. By Lemma 12, $g_{\Omega,R}(S|L') = \sum_{p\in\mathcal{I}_L} |E'_p|$ and $g_{\Omega,R}(S|L_0) = \sum_{p\in\mathcal{I}_L} |E^0_p|$. So if $\omega_{24} = 0$, $g_{\Omega,R}(S|L') = \omega_{11}|E'_{11}| + \omega_{12}|E'_{12}| + \omega_{23}|E'_{23}| + 0 \times |E'_{24}| = g_{\Omega,R}(S|L_0)$. Since G is a subgraph of G, $g_{\Omega,R}(S|L_0) = g_{\Omega,R}(S|G)$. □

Lemma 17. *Consider input tuple $(G(V,E),R,\Omega)$ with base algorithm $\mathcal{A}$ defined in Lemma 8: $f^T_{\mathcal{A}}(|G|)$ and $f^S_{\mathcal{A}}(|G|)$ denote the time and space of algorithm $\mathcal{A}$ on G respectively where $|G| \doteq |E|$. Denote by $V_0 = \mathcal{N}^+(R)$, $E_0 = E^+(R)$, graph $L_0(V_0,E_0) \subseteq G$ let $S_0 \subseteq V_0$ be the* LDS *on L_0. If Ω has $\omega_{24} = 0$ then (1) S_0 is the* LDS *on G, and (2)* ExpansionFramework *with $(G(V,E),R,\Omega)$ and $\mathcal{A}$ finds the* LDS *of G in $f^T_{\mathcal{A}}(\mathsf{vol}(R))$ time and $f^S_{\mathcal{A}}(\mathsf{vol}(R))$ space.*

Proof. We prove (1) by contradiction. Suppose that there is a subgraph $S' \neq S_0$ such that $\rho_{\Omega,R}(S'|G) = [\rho^*_{\Omega,R}|G] > \rho_{\Omega,R}(S_0|G)$. By Lemma 11, $\rho_{\Omega,R}(S'|G) \leq \rho_{\Omega,R}(S' \cap V_0|L_0)$, and since S_0 is the LDS on L_0, $\rho_{\Omega,R}(S' \cap V_0|L_0) \leq [\rho^*_{\Omega,R}|L_0] = \rho_{\Omega,R}(S_0|L_0)$. By Lemma 16, $\rho_{\Omega,R}(S' \cap V_0|L_0) = \rho_{\Omega,R}(S' \cap V_0|G)$ and $\rho_{\Omega,R}(S_0|L_0) = \rho_{\Omega,R}(S_0|G)$, so by Lemma 11, $\rho_{\Omega,R}(S'|G) \leq \rho_{\Omega,R}(S' \cap V_0|L_0) \leq \rho_{\Omega,R}(S_0|L_0) = \rho_{\Omega,R}(S_0|G)$, contradicting to the assumption $\rho_{\Omega,R}(S'|G) > \rho_{\Omega,R}(S_0|G)$. To see (2), $\mathcal{A}(L_0,R,\Omega)$ outputs the LDS S_0 of L_0 with $|L_0| = O(\mathsf{vol}(R))$ (Lemma 8), thus proves the lemma. □

Lemma 18. *Consider input tuple $(G(V,E),R,\Omega)$ where Ω has $\omega_{24} < 0$. Let S^* be the* LDS, *then for each $v \in S^*$, $\deg(v)$ is bounded by $(3 - \frac{2}{\omega_{24}})\mathsf{vol}(R)$. Let U_d be an alias of $(3 - \frac{2}{\omega_{24}})\mathsf{vol}(R)$, $\deg(v) = O(\mathsf{vol}(R)) = O(U_d)$.*

Proof. For a node $v \in S^*$, if $v \in R$, then $\deg(v) \leq \mathsf{vol}(R) \leq (3 - \frac{2}{\omega_{24}})\mathsf{vol}(R)$ as $\omega_{24} < 0$. Below, we consider a node $v \in S^* \setminus R$. Denote by $\{E_p|p \in \mathcal{I}\}$ the edge partitioning of $\mathcal{E}(S^*,R|G)$. By Lemma 12, $\rho_{\Omega,R}(S^*) = \frac{\sum_{p\in\mathcal{I}_L} \omega_p|E_p|}{|S^*|}$ where $\mathcal{I}_L = \{(1,1),(1,2),(2,3),(2,4)\}$. Thus, $(-\omega_{24})|E_{24}| = \omega_{11}|E_{11}| + \omega_{12}|E_{12}| + \omega_{23}|E_{23}| - \rho_{\Omega,R}(S^*)|S^*| \leq 2|E_{11}| + 2|E_{12}|$ because according to Definition 4, $\omega_{11} = 2, \omega_{12} \leq 2$ and $\omega_{23} \leq 0$. Since E_{11} are edges among nodes in $V_1 \doteq S^* \cap R$, and E_{12} are edges between V_1 and $V_2 \doteq S^* \setminus R$, E_{11} and E_{12} are two disjoint set of edges on V_1, so $|E_{12}|+|E_{12}| \leq \mathsf{vol}(V_1) \leq \mathsf{vol}(R)$. Thus, $(-\omega_{24})|E_{24}| \leq 2\mathsf{vol}(R)$. Because $\omega_{24} < 0$, $|E_{24}| \leq -\frac{2\mathsf{vol}(R)}{\omega_{24}}$. Because $v \in S^* \setminus R = V_2$, note that $R = V_1 \cup V_3$ where $V_3 \doteq R \setminus S^*$, any edge incident on v must be in one of the three sets $(\{v\} \times R) \cap E \subseteq (V_2 \times (V_1 \cup V_3)) \cap E = E_{12} \cup E_{23}$, $(\{v\} \times V_2) \cap E \subseteq E_{22}$ and E_{24}. Moreover, $|E \cap (\{v\} \times R)| \leq \mathsf{vol}(R)$ and $|(\{v\} \times V_2) \cap E| \leq |V_2| \leq |S^*| \leq 2\mathsf{vol}(R)$ according to Lemma 15. Thus, $\deg(v) \leq \mathsf{vol}(R) + 2\mathsf{vol}(R) + |E_{24}| \leq 3\mathsf{vol}(R) - \frac{2\mathsf{vol}(R)}{\omega_{24}}$. Therefore, $\deg(v) \leq (3 - \frac{2}{\omega_{24}})\mathsf{vol}(R)$. □

Recall that Lemma 15 defines $U_s \doteq 2\mathsf{vol}(R)$ and Lemma 18 defines $U_d \doteq (3 - \frac{2}{\omega_{24}})\mathsf{vol}(R)$.

Lemma 19. *Apply* ExpansionFramework *on input tuple $(G(V,E),R,\Omega)$ and base algorithm $\mathcal{A}$ defined in Lemma 8. Consider two consecutive iterations, i and $i+1$, with working graphs $L_i(V_i,E_i)$*

and $L_{i+1}(V_{i+1}, E_{i+1})$. If Ω has $\omega_{24} < 0$, then the number of nodes and edges added to L_i to form L_{i+1} are bounded, i.e., $\max\{|V_{i+1}| - |V_i|, |E_{i+1}| - |E_i|\} \leq U_s U_d = 2(3 - \frac{2}{\omega_{24}})\mathrm{vol}^2(R) = O(\mathrm{vol}^2(R))$. Let U_g be an alias for $U_s U_d$, both $|V_{i+1}| - |V_i|$ and $|E_{i+1}| - |E_i|$ are bounded by U_g.

Proof. Since $C_{i+1} = C_i \cup S_i$, $V_i = \mathcal{N}^+(C_i)$ and $V_{i+1} = \mathcal{N}^+(C_{i+1})$, the added nodes $V_{i+1} \setminus V_i$ are the neighbors of S_i. As $V_i \subseteq V_{i+1}$, $|V_{i+1}| - |V_i| = |V_{i+1} \setminus V_i| \leq |S_i| \times U_d$ where U_d is the upper bound of degree of nodes in S_i (Lemma 18). Note that $|S_i| \leq 2\mathrm{vol}(R) = U_s$ by Lemma 15. $|V_{i+1}| - |V_i| \leq U_s U_d$.

Since $E_i = E^+(C_i) = (C_i \times V) \cap E$ and $C_i \subseteq C_{i+1}$, $E_{i+1} = E^+(C_{i+1}) = (C_{i+1} \times V) \cap E = ((C_i \cup (C_{i+1} \setminus C_i)) \times V) \cap E = ((C_i \times V) \cap E) \cup (((C_{i+1} \setminus C_i) \times V) \cap E) = E_i \cup E^+(C_{i+1} \setminus C_i)$. Thus, $|E_{i+1}| - |E_i| \leq |E^+(C_{i+1} \setminus C_i)| \leq \mathrm{vol}(C_{i+1} \setminus C_i)$. Because $C_{i+1} \setminus C_i \subseteq S_i$, $|E_{i+1}| - |E_i| \leq \mathrm{vol}(S_i) \leq |S_i| U_d \leq U_s U_d$. □

Lemma 20. *Apply* ExpansionFramework *on input tuple (G, R, Ω) and a base algorithm $\mathcal{A}$ defined in Lemma 8. Consider two consecutive iterations i and $i+1$. Denote by $L_i(V_i, E_i)$ and $L_{i+1}(V_{i+1}, E_{i+1})$ the working graphs and S_i and S_{i+1} the* LDS *s found by $\mathcal{A}$, resp.. If Ω has $\omega_{24} < 0$, then*

$$\begin{cases} \rho_{\Omega,R}(S_i|L_i) > \rho_{\Omega,R}(S_{i+1}|L_{i+1}), & S_{i+1} \setminus V_i \neq \emptyset; \\ \rho_{\Omega,R}(S_i|L_i) \geq \rho_{\Omega,R}(S_{i+1}|L_{i+1}), & \textit{otherwise}. \end{cases}$$

Furthermore, when $\rho_{\Omega,R}(S_i|L_i) = \rho_{\Omega,R}(S_{i+1}|L_{i+1})$, $\mathcal{N}^+(S_{i+1}) \subseteq V_{i+1}$. In other words, the local density $\rho_{\Omega,R}(S_i|L_i)$ strictly decreases across iterations; once the decreasing stops, ExpansionFramework *halts. (Proof in Appendix A.5)*

Recall $U_s \doteq 2\mathrm{vol}(R)$ defined in Lemma 15, $U_d \doteq \mathrm{vol}(R)(3 - \frac{2}{\omega_{24}})$ in Lemma 18, and $U_g \doteq U_s U_d$ in Lemma 19. We discuss the complexity based on whether the weights in Ω are integers.

Lemma 21. *Apply* ExpansionFramework *to input tuple $(G(V, E), R, \Omega)$ with base algorithm $\mathcal{A}$ defined in Lemma 8. Let k be the maximum value of i in* ExpansionFramework *before termination, i.e., the number of iterations is $k+1$. If Ω has $\omega_{24} < 0$, then if all the weights in Ω are integers, then $k \leq U_s^2$; otherwise, $k \leq (U_s U_d + 1)^4 U_s$. Define alias $U_{\mathbb{Z}} \doteq U_s^2$ and alias $U_{\mathbb{R}} \doteq (U_s U_d + 1)^4 U_s$, so $k \leq U_{\mathbb{Z}} = O(\mathrm{vol}^2(R))$ when the weights Ω are integers and $k \leq U_{\mathbb{R}} = O(\mathrm{vol}^9(R))$ otherwise. (Proof in Appendix A.6)*

Lemma 22. *Consider input tuple $(G(V, E), R, \Omega)$ with base algorithm $\mathcal{A}$ defined in Lemma 8. Denote by k the largest value of i in the iteration of* ExpansionFramework. *If Ω has $\omega_{24} < 0$ then*

(1) If the weights in Ω are all integers, $|E_k| \leq U_g U_{\mathbb{Z}} = O(\mathrm{vol}^4(R))$;

(2) Otherwise, $|E_k| \leq U_g U_{\mathbb{R}} = O(\mathrm{vol}^{11}(R))$.

Proof. As the working graph can have, per iteration l before the termination (i.e., $0 \leq l < k$), U_g more edges by Lemma 19 and k is bounded by $U_{\mathbb{Z}}$ if the weights in Ω are all integers or $U_{\mathbb{R}}$ otherwise by Lemma 21, the number of edges of the last working graph is thus $|E_k| \leq k \times U_g$ which is bounded by $U_g U_{\mathbb{Z}} = O(\mathrm{vol}^4(R))$ when Ω are integers and by $U_g U_{\mathbb{R}} = O(\mathrm{vol}^{11}(R))$ otherwise. □

Lemma 23. *Consider input tuple $(G(V, E), R, \Omega)$ with $\mathcal{A}$ defined in Lemma 8. If Ω has $\omega_{24} < 0$, then the* LDS *S^* of G can be computed with a strongly local algorithm. Specifically,*

1. *If the weights in Ω are all integers, S^* can be computed in time $O(\mathrm{vol}^2(R) f^T_{\mathcal{A}}(\mathrm{vol}^4(R)))$ and space of $O(f^S_{\mathcal{A}}(\mathrm{vol}^4(R)))$;*
2. *Otherwise, S^* can be computed in $O(\mathrm{vol}^9(R) f^T_{\mathcal{A}}(\mathrm{vol}^{11}(R)))$ time and $O(f^S_{\mathcal{A}}(\mathrm{vol}^{11}(R))$ space.*

Proof. The size, i.e., the number of edges of the last (the largest) working subgraph shown in Lemma 15, bounds the space consumption, which is $O(\mathrm{vol}^4(R))$ when Ω consists of integer weights and $O(\mathrm{vol}^{11}(R))$ in other cases. Each call of $\mathcal{A}$ on a working graph $L_l(V_l, E_l)$ in iteration $0 \leq l \leq k$ takes $f^T_{\mathcal{A}}(|L_l|) = f^T_{\mathcal{A}}(|E_l|)$ time where $|E_l| \leq |E_k|$. The time consumption is thus bounded by the multiplication of the number of iterations provided by Lemma 21, i.e., $O(\text{vol}^2(R))$ when Ω have integer weights and $O(\text{vol}^9(R))$ in other cases, and $f^T_{\mathcal{A}}(|E_k|)$ where $|E_k|$ is bounded by Lemma 22. □

3 GENERALIZED LP-BASED SOLUTION

Table 1 shows that all the existing strongly local algorithms for local community search are flow-based. This paper provides a generic linear-programming-based strongly local approach for C_{LP}, configurations of C_L whose ω_{23} are 0.

Definition 9. $C_{LP} \doteq \{\Omega \in C_L | \omega_{23} = 0\}$.

This section considers an input tuple (G, R, Ω) with $\Omega \in C_{LP}$, proposes the LP formulation LP-DenLCS in Definition 12. Theorem 6 proves that LP-DenLCS can correctly find density-based LCS. Theorem 7 proves that by adding a constraint to LP-DenLCS, LP-DenLCS$^+$ correctly finds the maximal solution via binary search and thus can fit in ExpansionFramework with the analysis in Section 2 applied naturally.

Lemma 24. *Define $\mathcal{I}_{LP} \doteq \{(1,1),(1,2),(2,4)\}$. For any $\Omega \in C_{LP}$, $\omega_{11} = 2, \omega_{12} \geq 0, \omega_{24} \leq 0$, and for $\forall p \in \mathcal{I} \setminus \mathcal{I}_{LP}$, $\omega_p = 0$.*

Proof. $\omega_{11} = 2$, $\omega_{12} \geq 0$, $\omega_{24} \leq 0$ by Constraint **C1**, **C3** and **C4** respectively. For $p \in \mathcal{I} \setminus \mathcal{I}_L$, $\omega_p = 0$ by Lemma 12, besides, $\omega_{23} = 0$, thus $\omega_p = 0$ for $p \in \mathcal{I} \setminus \mathcal{I}_{LP}$. □

3.1 Generic LP Formulation

In order to introduce our LP formulation, we first define a partition of the edges in E based on the number of nodes an edge has in R or S. This partitioning is coarser than $\mathcal{E}(S, R|G)$ in Definition 2.

Definition 10. *Given an input tuple $(G(V,E), R, \Omega)$, define the following edge partitioning.*

(1) For k in $\{0,1,2\}$, $E_k(R)$ is the set of edges in E where each edge e has exactly k nodes in R, i.e., $e \in E_{|e\cap R|}(R)$. Formally, $E_0(R) = (\overline{R} \times \overline{R}) \cap E$, $E_1(R) = (R \times \overline{R}) \cap E$, and $E_2(R) = (R \times R) \cap E$. Denote by $\mathcal{E}(R|G) \doteq \{E_k(R) | k = 0,1,2\}$ the above edge partitioning based on R.

(2) For k in $\{0,1,2\}$ and any $S \subseteq V$, $E_k(S)$ is the set of edges in E where each edge e has k nodes in S, i.e., $e \in E_{|e\cap S|}(S)$. Formally, $E_0(S) = (\overline{S} \times \overline{S}) \cap E$, $E_1(R) = (S \times \overline{S}) \cap E$, and $E_2(S) = (S \times S) \cap E$. Denote by $\mathcal{E}(S|G) \doteq \{E_k(S) | k = 0,1,2\}$ the above edge partitioning based on S.

We now show that the edge partitionings of Definition 10 are coarsening the one in Definition 2.

Lemma 25. *Consider an input tuple $(G(V,E), R, \Omega)$ and a node set $S \subseteq V$. Denote by $\{E_p | p \in \mathcal{I}\} = \mathcal{E}(S,R)$ the edge partitioning of Definition 2 and by $\{E_k(R) | k = 0,1,2\} = \mathcal{E}(R)$ and $\{E_k(S) | k = 0,1,2\} = \mathcal{E}(S)$ that in Definition 10. Let (1) $\mathcal{I}_0(R) \doteq \{2,4\} \times \{2,4\}$, $\mathcal{I}_1(R) \doteq \{1,3\} \times \{2,4\}$, $\mathcal{I}_2(R) \doteq \{1,3\} \times \{1,3\}$, and (2) $\mathcal{I}_0(S) \doteq \{3,4\} \times \{3,4\}$, $\mathcal{I}_1(S) \doteq \{1,2\} \times \{3,4\}$, $\mathcal{I}_2(S) \doteq \{1,2\} \times \{1,2\}$, then*

(1) $E_0(R) = \cup_{p\in\mathcal{I}_0(R)} E_p$, $E_1(R) = \cup_{p\in\mathcal{I}_1(R)} E_p$, $E_2(R) = \cup_{p\in\mathcal{I}_2(R)} E_p$;

(2) $E_0(S) = \cup_{p\in\mathcal{I}_0(S)} E_p$, $E_1(S) = \cup_{p\in\mathcal{I}_1(S)} E_p$, $E_2(S) = \cup_{p\in\mathcal{I}_2(S)} E_p$.

Proof. Recall that the node partitioning $\mathcal{V}(S,R)$ on G, as defined in Definition 2, is (V_1, V_2, V_3, V_4) where $V_1 \doteq S \cap R$, $V_2 \doteq S \setminus R$, $V_3 \doteq R \setminus S$ and $V_4 \doteq \overline{R \cup S}$. Thus $R = V_1 \cup V_3$ while $\overline{R} = V_2 \cup V_4$, and $S = V_1 \cup V_2$ while $\overline{S} = V_3 \cup V_4$. Therefore, $E_0(R) = (\overline{R} \times \overline{R}) \cap E = ((V_2 \cup V_4) \times (V_2 \cup V_4)) \cap E = ((V_2 \times V_2) \cap E) \cup ((V_2 \times V_4) \cap E) \cup ((V_4 \times V_4) \cap E) = E_{22} \cup E_{24} \cup E_{44} = \cup_{p\in\mathcal{I}_0(R)=\{2,4\}\times\{2,4\}} E_p$. Similarly, $E_1(R) = (R \times \overline{R}) \cap E = ((V_1 \cup V_3) \times (V_2 \cup V_4)) \cap E$, Therefore, $E_1(R) = \cup_{p\in\{1,3\}\times\{2,4\}} E_p$. Similarly, $E_2(R) = (R \times R) \cap E$, since $R = (V_1 \cup V_3)$, $E_2(R) = \cup_{p\in\{1,3\}\times\{1,3\}} E_p$. The above proof to $\mathcal{E}(R)$ symmetrically applies to $\mathcal{E}(S)$. □

Now we define effective pair sets and a function for the LP formulation.

Definition 11. *Given input tuple $(G(V,E),R,\Omega)$ and a node set $S \subseteq V$, denote by $\{E_p | p \in \mathcal{I}\} = \mathcal{E}(S,R)$ the edge partitioning of Definition 2. For k in $\{0,1,2\}$, consider $\mathcal{I}_k(R)$ in Lemma 25, define*

- ***D1*** *Effective pair set $C_k \doteq \{p \in \mathcal{I}_{LP} \cap \mathcal{I}_k(R) | \Omega(p) \neq 0,\ E_p \neq \emptyset\}$,*
- ***D2*** *Effective weight λ_k, the weight of $\Omega(p)$ for $p \in C_k$ with the highest absolute value (break ties arbitrarily), formally defined in Eqn 6,*
- ***D3*** *Trident function for each edge $e \in E$ and $\forall a,b,c \in \{-1,0,1\}$ as Eqn 7, which takes a value in $\{a,b,c\}$ based on the number of nodes e has in R and the weights in Ω.*

$$\lambda_k = \begin{cases} \Omega(\arg\max_{p \in C_k} |\Omega(p)|), & C_k \neq \emptyset; \\ 0 & \textit{otherwise.} \end{cases} \quad (6) \qquad \Gamma_{a,b,c}(e) = \begin{cases} a & \textit{if } \lambda_{|e \cap R|} > 0, \\ b & \textit{if } \lambda_{|e \cap R|} = 0, \\ c & \textit{if } \lambda_{|e \cap R|} < 0. \end{cases} \quad (7)$$

With the above definitions, we are ready to introduce our LP formulation of density-based LCS.

Definition 12 (LP-DenLCS). *Given an input tuple $(G(V,E),R,\Omega)$ with $\Omega \in C_{LP}$, formulate the linear programming problem with variables $\boldsymbol{x} = \{x_v | v \in V\}$ and $\boldsymbol{y} = \{y_e | e \in E\}$ below.*

$$\boldsymbol{max} f_{\Omega,R}(\boldsymbol{x},\boldsymbol{y}) \doteq \sum_{k \in \{0,1,2\}} (\lambda_k \sum_{e \in E_k(R)} y_e) = \sum_{e \in E} \lambda_{|e \cap R|} \cdot y_e \tag{8}$$

$$\boldsymbol{s.t.}\ x_u \geq 0, \forall u \in V \tag{9}$$

$$y_e \geq 0, \forall e \in E \tag{10}$$

$$\|\boldsymbol{x}\| \doteq \sum_{u \in V} x_u \leq 1, \tag{11}$$

$$\Gamma_{1,1,-1}(e) y_e \leq \Gamma_{1,0,-1}(e) x_u + \Gamma_{0,0,1}(e) x_v, \forall e(u,v) \in E \tag{12}$$

$$\Gamma_{1,1,-1}(e) y_e \leq \Gamma_{1,0,-1}(e) x_v + \Gamma_{0,0,1}(e) x_u, \forall e(u,v) \in E \tag{13}$$

We now show that the assignment of $\mathbf{y}$ can be determined by that of $\mathbf{x}$ in optimizing $f_{\Omega,R}(\mathbf{x},\mathbf{y})$.

Lemma 26 (Optimal $\mathbf{y}$). *For any assignment of $\boldsymbol{x}$ satisfying Constraints (9) and (11) of LP-DenLCS, a feasible solution $(\boldsymbol{x},\boldsymbol{y})$ that maximizes $f_{\Omega,R}(\boldsymbol{x},\boldsymbol{y})$ can be derived with $\boldsymbol{y} = \{y_e | e \in E\}$ defined as*

$$\textit{For } \forall e(u,v) \in E, y_e = \begin{cases} \min\{x_u, x_v\} & \textit{if } \lambda_{|e \cap R|} > 0, \\ 0 & \textit{if } \lambda_{|e \cap R|} = 0, \\ |x_u - x_v| & \textit{if } \lambda_{|e \cap R|} < 0. \end{cases}$$

Proof. We show that any feasible solution $(\mathbf{x},\mathbf{y}')$ with $\mathbf{y}' = \{y'_e | e \in E\}$ has objective value $f_{\Omega,R}(\mathbf{x},\mathbf{y}') \leq f_{\Omega,R}(\mathbf{x},\mathbf{y})$ where $\mathbf{y}$ is defined above based on $\mathbf{x}$. Specifically, we perform the following process to each edge $e \in E$ iteratively to convert y'_e to y_e while maintaining the feasibility of the solution and non-strictly increasing the objective value, eventually having $f_{\Omega,R}(\mathbf{x},\mathbf{y}') \leq f_{\Omega,R}(\mathbf{x},\mathbf{y})$. Consider an edge $e \in E$ and Constraints 12-13 in LP-DenLCS. Recall the trident function $\Gamma_{a,b,c}(e)$ defined in Definition 10.

(1) If $\lambda_{|e \cap R|} > 0$, then $\Gamma_{1,1,-1}(e) = \Gamma_{1,0,-1}(e) = 1$ and $\Gamma_{0,0,1}(e) = 0$, so Constraint (12) becomes $y_e \leq x_u$ and Constraint (13) becomes $y_e \leq x_v$, and thus $y_e \in [0, \min\{x_u, x_v\}]$.
(2) If $\lambda_{|e \cap R|} = 0$, then $\Gamma_{1,1,-1}(e) = 1$ and both $\Gamma_{1,0,-1}(e) = \Gamma_{0,0,1}(e) = 0$, so $y_e \in [0,0]$, i.e., $y_e = 0$.
(3) If $\lambda_{|e \cap R|} < 0$, then $\Gamma_{1,0,-1} = \Gamma_{1,1,-1} = -1$ and $\Gamma_{0,0,1} = 1$, so $y_e \geq x_u - x_v$ and $y_e \geq x_v - x_u$, i.e., $y_e \geq |x_v - x_u|$.

Thus Constraints 12-13 can be summarized as (1) $0 \le y_e \le \min\{x_u, x_v\}$, if $\lambda_{|e\cap R|} > 0$; (2) $y_e = 0$, if $\lambda_{|e\cap R|} = 0$; (3) $y_e \ge |x_u - x_v|$, if $\lambda_{|e\cap R|} < 0$. Besides, $f_{\Omega,R}(\mathbf{x}, \mathbf{y}) = \sum_{e\in E} \lambda_{|e\cap R|} \cdot y_e$. When $\lambda_{|e\cap R|} > 0$, $f_{\Omega,R}$ increases with y_e, thus by increasing y_e to $\min\{x_u, x_v\}$, we non-strictly increase $f_{\Omega,R}$ while the solution is still feasible. When $\lambda_{|e\cap R|} = 0$, $y_e = 0$ is the only feasible assignment. When $\lambda_{|e\cap R|} < 0$, by decreasing y_e to $|x_u - x_v|$, we non-strictly increase $f_{\Omega,R}$ while the solution is still feasible. □

Lemma 26 suggests that with a feasible assignment of $\mathbf{x}$ we can determine a feasible assignment $\mathbf{y}$ that maximizes $f_{\Omega,R}(\mathbf{x}, \mathbf{y})$ under $\mathbf{x}$. Thus, in the following, we refer to a feasible solution **(x,y)** as $\mathbf{x}$ and the objective function as $f_{\Omega,R}(\mathbf{x}) \doteq f_{\Omega,R}(\mathbf{x}, \mathbf{y})$ since $\mathbf{y}$ can be determined by $\mathbf{x}$ via Lemma 26.

Throughout the rest of this section, we consider an input tuple $(G(V, E), R, \Omega)$ with $\Omega \in C_{LP}$ and $|E(R)| \ge 1$, so $\rho^*_{\Omega,R} \ge 1$ by Lemma 14.

Theorem 6. *Given an input tuple $(G(V, E), R, \Omega)$ with $\Omega \in C_{LP}$, let set S^* be* LDS, *i.e., $\rho_{\Omega,R}(S^*) = \rho^*_{\Omega,R}$, then the assignment $\boldsymbol{x}^*$ (note that $\boldsymbol{y}^*$ is determined by Lemma 26)*

$$x_u = \begin{cases} \frac{1}{|S^*|} & \text{if } u \in S^*; \\ 0 & \text{otherwise.} \end{cases}$$

is an optimal solution of LP-DenLCS *with the objective value $\rho_{\Omega,R}(S^*)$. (Proof in Appendix A.4)*

Theorem 7. *Given an input tuple $(G(V, E), R, \Omega)$ with $\Omega \in C_{LP}$, let $\rho^*_{\Omega,R}$ be the optimal value of* LP-DenLCS. *Adapt* LP-DenLCS *to parameterized* LP-DenLCS$^+$ *by adding an additional constraint of $x_u \le \frac{1}{k}, \forall u \in V$, where parameter k is a positive integer. Find the maximum integer k via a binary search over range $[1, |V|]$ such that* LP-DenLCS$^+$ *has the optimal value equal to $\rho^*_{\Omega,R}$; denote by $\boldsymbol{x}^*$ the assignment of the optimal solution. Such a solution exists because when $k = 1$,* LP-DenLCS$^+$ *is* LP-DenLCS. *Then, the set of nodes with non-zero assignment in $\boldsymbol{x}^*$ is a* LDS *for (G, R, Ω).*

Proof. Let the optimal value of LP-DenLCS$^+$ be ρ^+_k. As LP-DenLCS$^+$ is LP-DenLCS with an additional constraint, a feasible solution to LP-DenLCS$^+$ also applies to LP-DenLCS. Thus, $\rho^+_k \le \rho^*_{\Omega,R}$ for $\forall k \in [1, |V|]$. Denote by S^* the maximum DenLCS of input tuple (G, R, Ω) and $k^* = |S^*|$, then set assignment $\mathbf{x}^*$ via Theorem 7 is based on S^*, so $f_{\Omega,R}(\mathbf{x}^*) = \rho_{\Omega,R}(S^*) = \rho^*_{\Omega,R}$. Now we consider LP-DenLCS$^+$: if $k \le k^*$, $\mathbf{x}^*$ is a feasible with $f_{\Omega,R}(\mathbf{x}^*) = \rho^*_{\Omega,R}$, so $\rho^+_k = \rho^*_{\Omega,R}$; otherwise, by Lemma 34, for any solution $\mathbf{x}'$ with $\max_{x'_v\in\mathbf{x}'} x'_v < \frac{1}{k^*}$, $f_{\Omega,R}(\mathbf{x}') < \rho^*_{\Omega,R}$, so $\rho^+_k < \rho^*_{\Omega,R}$. Thus, for $\forall k$, by comparing ρ^+_k with $\rho^*_{\Omega,R}$, we can eventually reach k^* by updating k in each step of the binary search. □

Remarks. With Theorems 2-7, we draw a landscape of general density-based LCS. The general LP-based solution LP-DenLCS$^+$ can be plugged into the ExpansionFramework as $\mathcal{A}$ to form an efficient solution for LDS exploration under $\Omega \in C_{LP}$.

4 CONCLUSION

This paper introduces a broad class of density-based LCS objective functions. It provides a complete characterization of the parameter settings where a strongly local algorithm is possible. With one weight parameter set to 0, the paper provides a linear programming algorithm that is strongly local and practically efficient. Using the notion of strong locality, this paper characterizes a family of LCS problems while existing work only characterizes solutions for specific LCS problems.

ACKNOWLEDGMENTS

Miao Qiao was supported by Marsden Fund (UOA1732) and MBIE Catalyst: Strategic Fund NZ-Singapore Data Science Research Programme (UOAX2001). Rong-Hua Li was supported by the National Key Research and Development Program of China (2020AAA0108503) and NSFC Grants (U2241211 and 62072034).

REFERENCES

[1] Reid Andersen, Fan R. K. Chung, and Kevin J. Lang. 2006. Local Graph Partitioning using PageRank Vectors. In *47th Annual IEEE Symposium on Foundations of Computer Science (FOCS 2006), 21-24 October 2006, Berkeley, California, USA, Proceedings*. IEEE Computer Society, 475–486. https://doi.org/10.1109/FOCS.2006.44

[2] Yizhou Dai, Miao Qiao, and Lijun Chang. 2022. Anchored Densest Subgraph. In *SIGMOD '22: International Conference on Management of Data, Philadelphia, PA, USA, June 12 - 17, 2022*, Zachary G. Ives, Angela Bonifati, and Amr El Abbadi (Eds.). ACM, 1200–1213. https://doi.org/10.1145/3514221.3517890

[3] Maximilien Danisch, T.-H. Hubert Chan, and Mauro Sozio. 2017. Large Scale Density-friendly Graph Decomposition via Convex Programming. In *Proceedings of the 26th International Conference on World Wide Web, WWW 2017, Perth, Australia, April 3-7, 2017*, Rick Barrett, Rick Cummings, Eugene Agichtein, and Evgeniy Gabrilovich (Eds.). ACM, 233–242. https://doi.org/10.1145/3038912.3052619

[4] Adriano Fazzone, Tommaso Lanciano, Riccardo Denni, Charalampos E. Tsourakakis, and Francesco Bonchi. 2022. Discovering Polarization Niches via Dense Subgraphs with Attractors and Repulsers. *Proc. VLDB Endow.* 15, 13 (2022), 3883–3896. https://doi.org/10.14778/3565838.3565843

[5] Santo Fortunato and Mark E. J. Newman. 2022. 20 years of network community detection. *Nature Physics* 18, 8 (Aug. 2022), 848–850. https://doi.org/10.1038/s41567-022-01716-7

[6] Isabel M. Kloumann and Jon M. Kleinberg. 2014. Community membership identification from small seed sets. In *The 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '14, New York, NY, USA - August 24 - 27, 2014*, Sofus A. Macskassy, Claudia Perlich, Jure Leskovec, Wei Wang, and Rayid Ghani (Eds.). ACM, 1366–1375. https://doi.org/10.1145/2623330.2623621

[7] Dongsheng Luo, Yuchen Bian, Yaowei Yan, Xiao Liu, Jun Huan, and Xiang Zhang. 2020. Local Community Detection in Multiple Networks. In *KDD '20: The 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, CA, USA, August 23-27, 2020*, Rajesh Gupta, Yan Liu, Jiliang Tang, and B. Aditya Prakash (Eds.). ACM, 266–274. https://doi.org/10.1145/3394486.3403069

[8] Chenhao Ma, Reynold Cheng, Laks V. S. Lakshmanan, and Xiaolin Han. 2022. Finding Locally Densest Subgraphs: A Convex Programming Approach. *Proc. VLDB Endow.* 15, 11 (2022), 2719–2732. https://www.vldb.org/pvldb/vol15/p2719-ma.pdf

[9] Lorenzo Orecchia and Zeyuan Allen Zhu. 2014. Flow-Based Algorithms for Local Graph Clustering. In *Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2014, Portland, Oregon, USA, January 5-7, 2014*, Chandra Chekuri (Ed.). SIAM, 1267–1286. https://doi.org/10.1137/1.9781611973402.94

[10] Lu Qin, Rong-Hua Li, Lijun Chang, and Chengqi Zhang. 2015. Locally Densest Subgraph Discovery. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Sydney, NSW, Australia, August 10-13, 2015*, Longbing Cao, Chengqi Zhang, Thorsten Joachims, Geoffrey I. Webb, Dragos D. Margineantu, and Graham Williams (Eds.). ACM, 965–974. https://doi.org/10.1145/2783258.2783299

[11] M. Sozio and A. Gionis. 2010. The community-search problem and how to plan a successful cocktail party. In *Proc. of KDD'10*. 939–948.

[12] Charalampos E. Tsourakakis, Francesco Bonchi, Aristides Gionis, Francesco Gullo, and Maria A. Tsiarli. 2013. Denser than the densest subgraph: extracting optimal quasi-cliques with quality guarantees. In *The 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2013, Chicago, IL, USA, August 11-14, 2013*, Inderjit S. Dhillon, Yehuda Koren, Rayid Ghani, Ted E. Senator, Paul Bradley, Rajesh Parekh, Jingrui He, Robert L. Grossman, and Ramasamy Uthurusamy (Eds.). ACM, 104–112. https://doi.org/10.1145/2487575.2487645

[13] Nate Veldt, Austin R Benson, and Jon Kleinberg. 2020. Minimizing Localized Ratio Cut Objectives in Hypergraphs. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 1708–1718.

[14] Nate Veldt, David F. Gleich, and Michael W. Mahoney. 2016. A Simple and Strongly-Local Flow-Based Method for Cut Improvement. In *Proceedings of the 33nd International Conference on Machine Learning, ICML 2016, New York City, NY, USA, June 19-24, 2016 (JMLR Workshop and Conference Proceedings, Vol. 48)*, Maria-Florina Balcan and Kilian Q. Weinberger (Eds.). JMLR.org, 1938–1947. http://proceedings.mlr.press/v48/veldt16.html

[15] Nate Veldt, Christine Klymko, and David F Gleich. 2019. Flow-based local graph clustering with better seed set inclusion. In *Proceedings of the 2019 SIAM International Conference on Data Mining*. SIAM, 378–386.

A APPENDIX

A.1 Case Study

This section shows an interactive local community exploration using the linear programming algorithm introduced in Section 3. The underlying graph is a collaboration network dblp[4] where each node represents an author and each undirected edge denotes the co-authorship of two authors on a paper. R can be considered as an invitation list drafted for a workshop or a panel discussion. In Figure 9, the label of a node is in the form of "node degree-author name", nodes in grey show a possible R determined by a random process around a random author. To improve the invitation list, people invited should form a community local to R, i.e., the subgraph has a higher density biased to R. Based on Section 1.4, by tuning $x = \omega_{12} \in [0, 2]$ and $y = \omega_{24} \leq 0$ in a configuration, denoted as $\Omega_{x,y}$, one can optimize the corresponding objective function using our strongly local LP algorithm (Section 3), i.e., use $G, R, \Omega_{x,y}$ to produce a refined invitation list $S_{x,y}$. Figure 9 shows the results of different configurations in colored rectangles marked as $S_{x,y}$. Edges with only one node displayed are not shown. Consider the rules of thumb, i.e., increasing x expands the searching area, especially on the nodes outside R, while decreasing y penalizes high-degree nodes that are not in R, in 2 scenarios below.

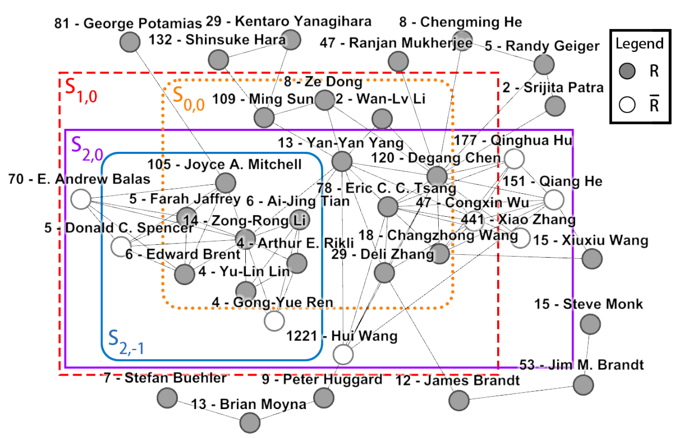


Fig. 9. Case Study: a Possible R and its LCS Results under 3 Configurations

Workshop. A workshop is expected to be open and inclusive. Fixing $y = 0$, a user can freely tweak x to decide to what extent people outside R are explored. For example, by choosing $x = 2$, the result $S_{2,0}$ contains more people outside of R, e.g., *Qiang He*, when compared with $S_{1,0}$. On the other hand, by having more people under consideration, $S_{2,0}$ has a higher density, e.g., $\rho(S_{1,0}) = 3.8$ and $\rho(S_{2,0}) = 5.1$, causing nodes with insufficient connections with other nodes to be excluded, such as *Ming Sun*. Further decreasing x to 0 produces a degenerated $S_{0,0}$ that only contains nodes in R.
Panel Discussion. A panel discussion prefers a more close-knit community; celebrities with light connections to the community may be excluded. Thus, $y = -1$ could be more suitable. Compared to $S_{2,0}$, $S_{2,-1}$ penalizes high-degree nodes that are not in R, and thus excludes *Andrew Balas* who has 65 out of 70 edges to $\overline{S_{2,-1}}$.

Note that the rules of thumb for tuning the two parameters can apply to other scenarios; moreover, due to the efficiency of our strongly local LP-based solution, exploration can be interactive. The user can not only choose the resulting communities but also explain the rationale behind the selection.

[4]http://konect.cc/networks/dblp_coauthor/

A.2 Proof of Lemma 1

For Equivalence (1), $k\frac{a}{b} > k\frac{a-c}{b-d} \iff k(ab-ad) > k(ab-bc) \iff kad < kbc \iff k\frac{a}{b} < k\frac{c}{d}$. For Equivalence (2), $k\frac{a}{b} > k\frac{a+c}{b+d} \iff k(ab+ad) > k(ab+bc) \iff kad > kbc \iff k\frac{a}{b} > k\frac{c}{d}$. The above proofs hold when substituting $<$ with $\leq$ and $>$ with $\geq$.

A.3 Proof of Lemma 2

Denote the node partitioning $\mathcal{V}(S,R)$ (Definition 2) of V as $\{V_1,V_2,V_3,V_4\}$. For each unordered pair $p(i,j) \in \mathcal{I}$, let $E_p = (V_i \times V_j) \cap E$. As $R = V_1 \cup V_3$ and $\overline{R} = V_2 \cup V_4$, $E^+(R) = (S \times V) \cap E = (S \times S \cup S \times \overline{S}) \cap E$, $E \setminus E^+(R) = (\overline{S} \times \overline{S}) \cap E = ((V_2 \cup V_4) \times (V_2 \cup V_4)) \cap E = E_{2,2} \cup E_{2,4} \cup E_{4,4}$. Based on the definition of C, for $\forall \Omega \in C$, we have $\omega_{44} = 0$ and $\omega_{24} \leq 0$; so if $\omega_{22} = 0$, for $\forall e \in E \setminus E^+(R)$ we have $w_{\Omega,R,S}(e) \leq \max\{\omega_{44}, \omega_{24}, \omega_{22}\} = 0$ according to Definition 3.

A.4 Proof of Theorem 6

We first introduce notations and then show the flow of the proof. For a solution $\mathbf{x} = \{x_v | v \in V\}$, we call $\|\mathbf{x}\| \doteq \sum_{u \in V} x_u$ the **sum** of $\mathbf{x}$, and $V^+(\mathbf{x}) = \{v \in V | x_v > 0\}$ the **key node set** of $\mathbf{x}$ which is the set of nodes with positive assignment. An assignment is **simple** if has at most one non-zero value, e.g., $\mathbf{x}$ is simple if $|\{x_v | v \in V\} \setminus \{0\}| = 1$ after de-duplication. To prove Theorem 6, Lemma 30 constructively proves that the lower bound of the optimal solution is $\rho_{\Omega,R}(S^*)$; Lemma 32 shows that there is always an optimal solution for LP-DenLCS that is simple; Lemma 33 shows that the upper bound of the simple optimal solution is $\rho_{\Omega,R}(S^*)$, which completes the proof of Theorem 6.

LEMMA 27 (SIMPLE y). *Given a simple feasible solution $\mathbf{x} = \{x_v | v \in V\}$, let $\{y_e | e \in E\}$ be the assignment of $\mathbf{y}$ based on Lemma 26, then $\mathbf{y}$ is simple. Specifically, if after de-duplication of the values, $\{x_v | v \in V\} = \{0, c\}$ then $\{y_e | e \in E\} = \{c, 0\}$.*

PROOF. Consider edge $e(u,v) \in E$. By Lemma 26, when $x_u, x_v \in \{0,c\}$, y_e also have $y_e \in \{0,c\}$ for all 3 cases of $\lambda_{|e \cap R|}$. □

Next, we discuss the general value of $\lambda_{|e\cap R|} \cdot y_e$ for $\forall e \in E$.

LEMMA 28. *Denote by $\{E_p | p \in \mathcal{I}\}$ the edge partitioning of $\mathcal{E}(S,R)$ an by $\mathbf{x} = \{x_v | v \in V\}$ a feasible solution of* LP-DenLCS. *Define $S \doteq \{v \in V | x_v = \max_{v' \in V} x_{v'}\}$. Then for $\forall e(u,v) \in E$ s.t. $x_u \geq x_v$,*

$$\lambda_{|e\cap R|} \cdot y_e = \begin{cases} \omega_{24}(x_u - x_v) & \text{if } e \in E_0(R); \\ \omega_{12} x_v & \text{if } e \in E_1(R); \\ \omega_{11} x_v & \text{if } e \in E_2(R). \end{cases}$$

PROOF. For value of λ_k where $k \in \{0,1,2\}$, from Definition 11(**D1**) $C_k = \{p \in \mathcal{I}_{LP} \cap \mathcal{I}_k(R) | \Omega(p) \neq 0,\ E_p \neq \emptyset\}$ and by Lemma 24, $\omega_{11} = 2, \omega_{12} \geq 0, \omega_{24} \leq 0$, and besides $\omega_p = 0$ for any $p \in \mathcal{I} \setminus \mathcal{I}_{LP}$.

- For $k = 0$, $\omega_{24} \leq 0, \omega_{44} = 0$ (Definition 4), $\omega_{22} = 0$ as $\Omega \in C_{LP}$, and by Lemma 25, $\mathcal{I}_0(R) = \{(2,2),(2,4),(4,4)\}$, so if $\omega_{24} < 0$ then $C_0 = \{(2,4)\}$, otherwise, $C_0 = \emptyset$. By Definition 11(**D2**) of λ_k, when $C_0 = \{(2,4)\}$, $\lambda_0 = \omega_{24}$, and otherwise, $\lambda_0 = 0 = \omega_{24}$. By Lemma 26, for any $e(u,v) \in E_0(R)$, because $\lambda_{|e\cap R|} = \lambda_0 = \omega_{24} \leq 0$, if $\omega_{24} < 0$ then $y_e = |x_u - x_v| = x_u - x_v$ and otherwise $y_e = 0$. Observe that in both cases, $\lambda_{|e\cap R|} \cdot y_e = \omega_{24}(x_u - x_v)$.
- For $k = 1$, $\omega_{12} \geq 0$ and $\omega_{34} = 0$ (Definition 4), $\omega_{14} = \omega_{23} = 0$ as $\Omega \in C_{LP}$. By Lemma 25, $\mathcal{I}_1(R) = \{(1,2),(1,4),(2,3),(3,4)\}$, so if $\omega_{12} > 0$ then $C_1 = \{(1,2)\}$, and if $\omega_{12} = 0$ then $C_1 = \emptyset$. By Definition 11(**D2**) of λ_k, when $C_1 = \{(1,2)\}$, $\lambda_1 = \omega_{12}$; when $C_1 = \emptyset$, $\lambda_1 = 0 = \omega_{12}$. By Lemma 26, for any $e(u,v) \in E_1(R)$, since $\lambda_{|e\cap R|} = \lambda_1 = \omega_{12} \geq 0$, it follows that $y_e = \min\{x_u, x_v\} = x_v$ if $\omega_{12} > 0$ and $y_e = 0$ otherwise, and in both cases, $\lambda_{|e\cap R|} \cdot y_e = \omega_{12} x_v$.

- For $k = 2$, $\omega_{11} = 2, \omega_{33} = 0, \omega_{14} \leq \omega_{13} \leq 0$ (Definition 4), since $\omega_{14} = 0$ for $\Omega \in C_{LP}$, we have $\omega_{13} = 0$. By Lemma 25, $\mathcal{I}_2(R) = \{(1,1),(1,3),(3,3)\}$, so $C_2 = \{(1,1)\}$. By Definition 11(**D2**), $\lambda_2 = \omega_{11}$. By Lemma 26, for $\forall e(u,v) \in E_2(R)$, since $\lambda_{|e\cap R|} = \lambda_2 = \omega_{11} > 0$, $y_e = \min\{x_u, x_v\} = x_v$ because $x_u \geq x_v$ and therefore $\lambda_{|e\cap R|} \cdot y_e = \omega_{11}x_v$. □

Lemma 29. *Given $S \subseteq V$, the assignment of $\boldsymbol{x}$ with $0 \leq c \leq \frac{1}{|S|}$ can be defined as $x_u = c$ if $u \in S$, and $x_u = 0$ otherwise, which is feasible with $f_{\Omega,R}(\boldsymbol{x}) = cg_{\Omega,R}(S) = c|S|\rho_{\Omega,R}(S) = \|\boldsymbol{x}\|\rho_{\Omega,R}(S) \leq \rho_{\Omega,R}(S)$.*

Proof. $\mathbf{x}$ is a feasible solution as $\|\mathbf{x}\| = \sum_{u\in V} x_u = |S|c \leq 1$ and all its values are non-negative. Consider an edge $e(u,v)$. W.l.o.g., we assume $x_u \geq x_v$, thus, if $|e \cap S| = 0$, then $x_u, x_v \notin S$, thus $x_u = x_v = 0$; If $|e \cap S| = 1$, then $x_u \in S$ and $x_v \notin S$, thus $x_u = c, x_v = 0$; And if $|e \cap S| = 2$, then $x_u, x_v \in S$, thus $x_u = x_v = c$. Denote by $t = |e \cap R|$, $e \in E_t(R)$ (Definition 10). Consider $\lambda_t \cdot y_e$:

$t = 0$. By Lemma 25, $E_0(R) = E_{22} \cup E_{24} \cup E_{44}$ where $E_{22} \subseteq E_2(S), E_{24} \subseteq E_1(S)$, and $E_{44} \subseteq E_0(S)$. Thus, if $e \in E_{22}$, then $|e \cap S| = 2$, $x_u - x_v = 0$, thus $\lambda_t y_e = \omega_{24}(x_u - x_v) = 0$ (Lemma 28); if $e \in E_{24}$, then $|e \cap S| = 1$, $x_u - x_v = c$, thus $\lambda_t y_e = \omega_{24}(x_u - x_v) = c\omega_{24}$(Lemma 28); if $e \in E_{44}$, then $|e \cap S| = 0$, thus $x_u - x_v = 0$, so $\lambda_t y_e = \omega_{24}(x_u - x_v) = 0$ (Lemma 28).

$t = 1$. By Lemma 25, $E_1(R) = E_{12} \cup E_{14} \cup E_{23} \cup E_{34}$ where $E_{12} \subseteq E_2(S), E_{14}, E_{23} \subseteq E_1(S), E_{34} \subseteq E_0(S)$. Thus, if $e \in E_{12}$, $|e \cap S| = 2$, $x_v = c$, by Lemma 28, $\lambda_t y_e = \omega_{12}x_v = \omega_{12}c$; if $e \in E_{14} \cup E_{23} \cup E_{34}$, then $|e \cap S| \leq 1$, $x_v = 0$, so $\lambda_t y_e = \omega_{12}x_v = 0$.

$t = 2$. By Lemma 25, $E_2(R) = E_{11} \cup E_{13} \cup E_{33}$, where $E_{11} \subseteq E_2(S), E_{13} \subseteq E_1(S), E_{33} \subseteq E_0(S)$. Thus, if $e \in E_{11}$, $|e \cap S| = 2$, $x_v = c$, by Lemma 28, $\lambda_t y_e = \omega_{11}x_v = \omega_{11}c$; if $e \in E_{13} \cup E_{33}$, $|e \cap S| \leq 1$, $x_v = 0$, so $\lambda_t y_e = \omega_{11}x_v = 0$ (Lemma 28).

To summarize the above discussion, $\lambda_{|e\cap R|} \cdot y_e = \omega_{11}c$ if $e \in E_{11}$, $\omega_{12}c$ if $e \in E_{12}$, $\omega_{24}c$ if $e \in E_{24}$, and 0 otherwise. Since for $\Omega \in C_{LP}$, $\omega_p = 0$ for $p \in \mathcal{I} \setminus \{(1,1),(1,2),(2,4)\}$, $f_{\Omega,R}(\mathbf{x}) = \omega_{11}c|E_{11}| + \omega_{12}c|E_{12}| + \omega_{24}c|E_{24}| = c\sum_{p\in\mathcal{I}} \omega_p|E_p| = cg_{\Omega,R}(S) = c|S|\rho_{\Omega,R}(S) = \|\mathbf{x}\|\rho_{\Omega,R}(S) \leq \rho_{\Omega,R}(S)$. □

Lemma 30 (Lower bound). *The optimal value of* LP-DenLCS $f^*_{\Omega,R} = \max_{\forall \text{ feasible } \boldsymbol{x}} f_{\Omega,R}(\boldsymbol{x}) \geq \rho^*_{\Omega,R}$.

Proof. Consider an LDS S^* with $\rho_{\Omega,R}(S^*) = \rho^*_{\Omega,R}$ and the corresponding $\mathbf{x}^*$ decided by Lemma 29 with $c = \frac{1}{|S^*|}$, thus $f_{\Omega,R}(\mathbf{x}^*) = \frac{1}{|S^*|}|S^*|\rho_{\Omega,R}(S^*) = \rho^*_{\Omega,R}$, proving the lemma. □

Definition 13 (Peeling). *Given a feasible solution $\boldsymbol{x} = \{x_v | v \in V\}$, define the $\boldsymbol{x}$-ordering of V as an non-increasing ordering of the nodes v in V under $\boldsymbol{x}$, i.e., $x_{v_1} \geq x_{v_2} \geq \ldots \geq x_{v_n}$, break ties arbitrarily. For each integer $i \in [n]$, denote by $\mathsf{Pre}(\boldsymbol{x}, i) \doteq \{v_1, v_2, \cdots, v_i\}$ the **prefix** of the $\boldsymbol{x}$-ordering under $\boldsymbol{x}$, let $\Delta(\boldsymbol{x}, i) \doteq x_{v_i} - x_{v_{i+1}} \geq 0$ be the difference between the assignments of v_i and v_{i+1}. Here $x_{v_{n+1}} = 0$ is a dummy value for the simplicity of formulation s.t. $\Delta(\boldsymbol{x}, n) = x_{v_n}$. Define the **peeling** of $\boldsymbol{x}$ as n solutions $\boldsymbol{x}^1, \boldsymbol{x}^2, \cdots \boldsymbol{x}^n$ of* LP-DenLCS *as follows: for $\forall l \in [1,n]$, $\boldsymbol{x}^l = \{x^l_{v_i} | i \in [n]\}$, where $x^l_{v_i} = \Delta(\boldsymbol{x}, l)$ if $i \leq l$, and $x^l_{v_i} = 0$ otherwise. Note that each prefix $pre(\boldsymbol{x}, i)$, $i \in [n]$ is a subgraph of G. Let $\rho^+_{\Omega,R}(\boldsymbol{x}) \doteq \max_{l\in[n]} \rho_{\Omega,R}(\mathsf{Pre}(\boldsymbol{x}, l))$ the maximum local density among all prefixes of $\boldsymbol{x}$-ordering. Let k be the maximum integer such that the k-prefix reaches density $\rho^+_{\Omega,R}(\boldsymbol{x}) = \rho_{\Omega,R}(\mathsf{Pre}(\boldsymbol{x}, k))$. k is called **the critical integer** of $\boldsymbol{x}$.*

Lemma 31 (Peeling property). *Given a feasible solution $\boldsymbol{x}$, let $\boldsymbol{x}^1, \cdots, \boldsymbol{x}^n$ be the peeling of $\boldsymbol{x}$. Then*

(1) For each $i \in [n]$, $\boldsymbol{x}^i$ is a simple feasible solution of LP-DenLCS,
(2) $\|\boldsymbol{x}\| = \sum_{i\in[n]} \|\boldsymbol{x}^i\|$, and
(3) $f_{\Omega,R}(\boldsymbol{x}) = \sum_{i\in[n]} f_{\Omega,R}(\boldsymbol{x}^i)$.

Proof. (1) Consider $i \in [n]$. $\mathbf{x}^i$ is simple as $\mathbf{x}^i = \{0, \Delta(\mathbf{x}, i)\}$. For $\forall v \in \mathsf{Pre}(\mathbf{x}, i)$, $x_{v_i} \leq x_v$, thus $0 \leq x^i_v = \Delta(\mathbf{x}, i) = x_{v_i} - x_{v_{i+1}} \leq x_{v_i} \leq x_v$. For $\forall v \notin \mathsf{Pre}(\mathbf{x}, i)$, $0 = x^i_v \leq x_v$. Thus $0 \leq \|x^i\| \leq \|x\| \leq 1$.

Thus, x^i is simple and feasible. (2) $\sum_{i\in[n]} \|\mathbf{x}^i\| = \sum_{i\in[n]} i \cdot \Delta(\mathbf{x}, i)$

$$= \sum_{i\in[n]} \sum_{j\in[1,i]} \Delta(\mathbf{x}, i) = \sum_{j\in[n]} \sum_{i\in[j,n]} \Delta(\mathbf{x}, i) \tag{14}$$

$$= \sum_{j\in[n]} (x_{v_j} - x_{v_{j+1}} + x_{v_{j+1}} - x_{v_{j+2}} + \cdots + x_{v_n} - x_{v_{n+1}}) = \sum_{j\in[n]} (x_{v_j} - x_{v_{n+1}}) = \sum_{j\in[n]} x_{v_j} = \|\mathbf{x}\|. \tag{15}$$

(3) For each integer $j \in [n]$, from Equations 14-15,

$$\sum_{i\in[j,n]} \Delta(\mathbf{x}, i) = x_{v_j}. \tag{16}$$

Define n solutions $\overline{\mathbf{x}}^1, \overline{\mathbf{x}}^2, \cdots, \overline{\mathbf{x}}^n$: For $\forall l \in [n], \overline{\mathbf{x}}^l = \{\overline{x}^l_{v_i} | i \in [n]\}$, where $\overline{x}^l_{v_i} = x_{v_l}$ if $i \le l$, and $\overline{x}^l_{v_i} = x_{v_i}$ otherwise. From Equation 16, $x_{v_l} = \sum_{j\in[l,n]} \Delta(\mathbf{x}, j)$ and $x_{v_i} = \sum_{j\in[i,n]} \Delta(\mathbf{x}, j)$. Note that each $\overline{\mathbf{x}}^l$ is feasible because when $i \le l$, $\overline{x}^l_{v_i} = x_{v_l} \le x_{v_i}$, when $i > l$, $\overline{x}^l_{v_i} = x_{v_i}$. Thus, $\overline{x}^l_{v_i} \le x^l_{v_i}$ for each $i \in [n]$ and thus $\|\overline{\mathbf{x}}^l\| \le \|\mathbf{x}\| = 1$. Besides, we can verify that $\overline{\mathbf{x}}^1 = \mathbf{x}$ and $\overline{\mathbf{x}}^n = \mathbf{x}^n$.

We now use induction starting from $l = n$ back to 1 to prove that $f_{\Omega,R}(\overline{\mathbf{x}}^l) = \sum_{i\in[l,n]} f_{\Omega,R}(\mathbf{x}^i)$ holds for all $l \in [n]$, which, when $l = 1$ proves (3) $f_{\Omega,R}(\mathbf{x}) = \sum_{i\in[n]} f_{\Omega,R}(\mathbf{x}^i)$.

When $l = n$, $f_{\Omega,R}(\overline{\mathbf{x}}^n) = f_{\Omega,R}(\mathbf{x}^n) = \sum_{i\in[n,n]} f_{\Omega,R}(\mathbf{x}^i)$. Assume that for $k > 2$, $f_{\Omega,R}(\overline{\mathbf{x}}^l) = \sum_{i\in[l,n]} f_{\Omega,R}(\mathbf{x}^i)$ for $l \in [k, n]$. We now prove that $f_{\Omega,R}(\overline{\mathbf{x}}^{k-1}) = \sum_{i\in[k-1,n]} f_{\Omega,R}(\mathbf{x}^i)$.

Consider $f_{\Omega,R}(\overline{\mathbf{x}}^{k-1}) - f_{\Omega,R}(\overline{\mathbf{x}}^k)$. Denote by: (1) $S = \mathrm{Pre}(\mathbf{x}, k-1) = \{v_1, v_2, \cdots, v_{k-1}\}$; (2) $\{E_p | p \in \mathcal{I}\}$, the edge partitioning of $\mathcal{E}(S, R)$; (3) $\overline{\mathbf{y}}^i = \{\overline{y}^i_e | e \in E\}$, obtained by applying Lemma 26 on $\overline{\mathbf{x}}^i$; (4) $\overline{\mathbf{y}}^{i-1} = \{\overline{y}^{i-1}_e | e \in E\}$, obtained by applying Lemma 26 on $\overline{\mathbf{x}}^{i-1}$. By Equation (8), $f_{\Omega,R}(\overline{\mathbf{x}}^{k-1}) = \sum_{e\in E} \lambda_{|e\cap R|} \overline{y}^{k-1}_e$ and $f_{\Omega,R}(\overline{\mathbf{x}}^k) = \sum_{e\in E} \lambda_{|e\cap R|} \overline{y}^k_e$. Recall that $S = \mathrm{Pre}(\mathbf{x}, k-1)$, so for any $i \in [n]$, if $i \le k-1$, then $v_i \in S$ and $\overline{x}^{k-1}_{v_i} = x_{v_{k-1}}$, and $\overline{x}^k_{v_i} = x_{v_k}$. If $i \ge k$, then $v_i \notin S$ and by definition, $\overline{x}^{k-1}_{v_i} = \overline{x}^k_{v_i} = x_{v_i}$. Consider edge $e(u, v)$. W.l.o.g., we assume u is before v in the $\mathbf{x}$-ordering, i.e., $x_u \ge x_v$, the values of $\overline{x}^{k-1}_u, \overline{x}^{k-1}_v, \overline{x}^k_u$ and $\overline{x}^k_v$ have the following properties.

- If $|e \cap S| = 0$, $u, v \notin S$, thus $\overline{x}^{k-1}_u = \overline{x}^k_u = x_u \ge x_v = \overline{x}^{k-1}_v = \overline{x}^k_v$
- If $|e \cap S| = 1$, $u \in S, v \notin S$, thus $\overline{x}^{k-1}_u = x_{v_{k-1}}, \overline{x}^k_u = x_{v_k}, \overline{x}^{k-1}_v = \overline{x}^k_v = x_v$, and thus $\overline{x}^{k-1}_u \ge \overline{x}^{k-1}_v, \overline{x}^k_u \ge \overline{x}^k_v$.
- If $|e \cap S| = 2$, $u, v \in S$, thus $\overline{x}^{k-1}_u = \overline{x}^{k-1}_v = x_{v_{k-1}}, \overline{x}^k_u = \overline{x}^k_v = x_{v_k}$, and thus $\overline{x}^{k-1}_u \ge \overline{x}^{k-1}_v, \overline{x}^k_u \ge \overline{x}^k_v$.

Note that, the above discussion shows that when $x_u \ge x_v$, $\overline{x}^{k-1}_u \ge \overline{x}^{k-1}_v$ and $\overline{x}^k_u \ge \overline{x}^k_v$, and thus we can apply Lemma 28 in the discussions below on the value of $\lambda_t(\overline{y}^{k-1}_e - \overline{y}^k_e)$ where $t = |e \cap R|$.

$t = 0$. By Lemma 28, $\lambda_t \overline{y}^{k-1}_e = \omega_{24}(\overline{x}^{k-1}_u - \overline{x}^{k-1}_v)$ and $\lambda_t \overline{y}^k_e = \omega_{24}(\overline{x}^k_u - \overline{x}^k_v)$, so $\lambda_t(\overline{y}^{k-1}_e - \overline{y}^k_e) = \omega_{24}((\overline{x}^{k-1}_u - \overline{x}^{k-1}_v) - (\overline{x}^k_u - \overline{x}^k_v))$. By Lemma 25, $E_0(R) = E_{22} \cup E_{24} \cup E_{44}$ where $E_{22} \subseteq E_2(S), E_{24} \subseteq E_1(S), E_{44} \subseteq E_0(S)$. Therefore,

- If $e \in E_{22}$, $|e \cap S| = 2$, $\overline{x}^{k-1}_u = \overline{x}^{k-1}_v = x_{v_{k-1}}, \overline{x}^k_u = \overline{x}^k_v = x_{v_k}$, so $\lambda_t(\overline{y}^{k-1}_e - \overline{y}^k_e) = \omega_{24}(0-0) = 0$;
- If $e \in E_{24}$, $|e \cap S| = 1$, $\overline{x}^{k-1}_u = x_{v_{k-1}}, \overline{x}^k_u = x_{v_k}, \overline{x}^{k-1}_v = \overline{x}^k_v$, so $\lambda_t(\overline{y}^{k-1}_e - \overline{y}^k_e) = \omega_{24}((x_{v_{k-1}} - \overline{x}^{k-1}_v) - (x_{v_k} - \overline{x}^{k-1}_v)) = \omega_{24}((x_{v_{k-1}} - x_{v_k}) - (\overline{x}^{k-1}_v - \overline{x}^k_v)) = \omega_{24}\Delta(\mathbf{x}, k-1)$;
- If $e \in E_{44}$, $|e \cap S| = 0$, $\overline{x}^{k-1}_u = \overline{x}^k_u, \overline{x}^{k-1}_v = \overline{x}^k_v$, so $\lambda_t(\overline{y}^{k-1}_e - \overline{y}^k_e) = \omega_{24}((\overline{x}^{k-1}_u - \overline{x}^{k-1}_v) - (\overline{x}^k_u - \overline{x}^k_v)) = \omega_{24}((\overline{x}^{k-1}_u - \overline{x}^k_u) - (\overline{x}^{k-1}_v - \overline{x}^k_v)) = 0$.

$t = 1$. By Lemma 28, $\lambda_t \overline{y}^{k-1}_e = \omega_{12} \overline{x}^{k-1}_v$ and $\lambda_t \overline{y}^k_e = \omega_{12} \overline{x}^k_v$. By Lemma 25, $E_1(R) = E_{12} \cup E_{14} \cup E_{23} \cup E_{34}$ where $E_{12} \subseteq E_2(S), E_{14}, E_{23} \subseteq E_1(S)$ and $E_{34} \subseteq E_0(S)$. Therefore,

- If $e \in E_{12}$, $|e \cap S| = 2$, $\overline{x}^{k-1}_v = x_{v_{k-1}}, \overline{x}^k_v = x_{v_k}$, so $\lambda_t(\overline{y}^{k-1}_e - \overline{y}^k_e) = \omega_{12}(\overline{x}^{k-1}_v - \overline{x}^k_v) = \omega_{12}(x_{v_{k-1}} - x_{v_k}) = \omega_{12}\Delta(\mathbf{x}, k-1)$;
- If $e \in E_{14} \cup E_{23} \cup E_{34}$, $|e \cap S| \le 1$, $\overline{x}^{k-1}_v = \overline{x}^k_v$, so $\lambda_t(\overline{y}^{k-1}_e - \overline{y}^k_e) = \omega_{12}(\overline{x}^{k-1}_v - \overline{x}^k_v) = 0$.

$t = 2$. By Lemma 28, $\lambda_t \overline{y}_e^{k-1} = \omega_{11}\overline{x}_v^{k-1}$ and $\lambda_t \overline{y}_e^{k} = \omega_{11}\overline{x}_v^{k}$. By Lemma 25, $E_2(R) = E_{11} \cup E_{13} \cup E_{33}$ where $E_{11} \subseteq E_2(S), E_{13} \subseteq E_1(S)$ and $E_{33} \subseteq E_0(S)$. Therefore,

- If $e \in E_{11}$, $|e \cap S| = 2$, $\overline{x}_v^{k-1} = x_{v_{k-1}}$, $\overline{x}_v^{k} = x_{v_k}$, so $\lambda_t(\overline{y}_e^{k-1} - \overline{y}_e^{k}) = \omega_{11}(\overline{x}_v^{k-1} - \overline{x}_v^{k}) = \omega_{11}(x_{v_{k-1}} - x_{v_k}) = \omega_{11}\Delta(\mathbf{x}, k-1)$;
- If $e \in E_{13} \cup E_{33}$, $|e \cap S| \leq 1$, $\overline{x}_v^{k-1} = \overline{x}_v^{k}$, so $\lambda_t(\overline{y}_e^{k-1} - \overline{y}_e^{k}) = \omega_{11}(\overline{x}_v^{k-1} - \overline{x}_v^{k}) = 0$.

From the above discussion, $\lambda_t(\overline{y}_e^{k-1} - \overline{y}_e^{k})$ equals $\omega_{11}\Delta(\mathbf{x}, k-1)$ if $e \in E_{11}$, $\omega_{12}\Delta(\mathbf{x}, k-1)$ if $e \in E_{12}$, $\omega_{24}\Delta(\mathbf{x}, k-1)$ if $e \in E_{24}$, and 0 otherwise. Since for $\Omega \in C_{LP}$, $\omega_p = 0$ for $p \in \mathcal{I} \setminus \{(1,1),(1,2),(2,4)\}$, we summarize that $f_{\Omega,R}(\overline{\mathbf{x}}^{k-1}) - f_{\Omega,R}(\overline{\mathbf{x}}^{k}) = \sum_{e \in E} \lambda_{|e \cap R|}(\overline{y}_e^{k-1} - \overline{y}_e^{k}) = \sum_{p \in \mathcal{I}_{LP}} \omega_p \Delta(\mathbf{x}, k-1)|E_p| = \Delta(\mathbf{x}, k-1)g_{\Omega,R}(S) = f_{\Omega,R}(\mathbf{x}^{k-1})$. Note the last equality comes from Lemma 29. This, alongside the inductive assumption, proves that $f_{\Omega,R}(\overline{\mathbf{x}}^{l}) = \sum_{i \in [l,n]} f_{\Omega,R}(\mathbf{x}^i)$ for $l = k-1$ and thus proves the correctness for any $l \in [n]$ whose special case when $l = 1$ is $f_{\Omega,R}(\mathbf{x}) = \sum_{i \in [n]} f_{\Omega,R}(\mathbf{x}^i)$. □

Lemma 32 (Merging property). *Given a feasible solution $\mathbf{x}$, let $\mathbf{x}^1, \mathbf{x}^2, \cdots, \mathbf{x}^n$ be the peeling of $\mathbf{x}$. Let k be the critical integer of $\mathbf{x}$ (Definition 13). Define a simple feasible solution $\mathbf{x}^+ = \{x_v^+ | v \in V\}$, where $x_v^+ = \frac{\|\mathbf{x}\|}{k}$ if $v \in \mathsf{Pre}(\mathbf{x}, k)$, and $x_v^+ = 0$ otherwise. Then we have $\|\mathbf{x}^+\| = \|\mathbf{x}\|$ and $f_{\Omega,R}(\mathbf{x}^+) \geq f_{\Omega,R}(\mathbf{x})$.*

Proof. $\|\mathbf{x}^+\| = \|\mathbf{x}\|$ because by definition, $\mathbf{x}^+$ has k values with $\frac{\|\mathbf{x}\|}{k}$ and 0 otherwise, so $\|\mathbf{x}^+\| = k\frac{\|\mathbf{x}\|}{k} = \|\mathbf{x}\|$. Also $\rho_{\Omega,R}(\mathsf{Pre}(\mathbf{x}, k)) \geq \rho_{\Omega,R}(\mathsf{Pre}(\mathbf{x}, i))$ for any $i \in [n]$ as k is critical. Thus,

$$\begin{aligned} f_{\Omega,R}(\mathbf{x}^+) &= \|\mathbf{x}^+\|\rho_{\Omega,R}(\mathsf{Pre}(\mathbf{x}, k)) \quad \text{(Lemma 29)} \\ &= \|\mathbf{x}\|\rho_{\Omega,R}(\mathsf{Pre}(\mathbf{x}, k)) = \sum_{i \in [n]} \|\mathbf{x}^i\|\rho_{\Omega,R}(\mathsf{Pre}(\mathbf{x}, k)) \quad \text{(Lemma 31 (2))} \\ &\geq \sum_{i \in [n]} \|\mathbf{x}^i\|\rho_{\Omega,R}(\mathsf{Pre}(\mathbf{x}, i)) = \sum_{i \in [n]} f_{\Omega,R}(\mathbf{x}^i) \quad \text{(Lemma 29)} \\ &= f_{\Omega,R}(\mathbf{x}) \quad \text{(Lemma 31 (3))} \end{aligned}$$

□

Lemma 33. *$f_{\Omega,R}(\mathbf{x}) \leq \rho^*_{\Omega,R}$ for any feasible solution $\mathbf{x}$.*

Proof. By Lemma 32, for any feasible solution $\mathbf{x}$, there exists a simple feasible solution $\mathbf{x}'$ s.t. $f_{\Omega,R}(\mathbf{x}') \geq f_{\Omega,R}(\mathbf{x})$. Let $S' = V^+(\mathbf{x}')$, the set of nodes with positive assignments in $\mathbf{x}'$, and thus $c = \frac{\|x\|}{|S'|}$. By Lemma 29, $f_{\Omega,R}(\mathbf{x}) \leq f_{\Omega,R}(\mathbf{x}') = c|S'|\rho_{\Omega,R}(S') = \|\mathbf{x}\|\rho_{\Omega,R}(S') \leq \rho_{\Omega,R}(S') \leq \rho^*_{\Omega,R}$. □

Lemma 34. *Let S^* be the maximal densest subgraph. Let $\mathbf{x} = \{x_v | v \in V\}$ be a feasible solution of* LP-DenLCS. *If $\max\{x_v | v \in V\} < \frac{1}{|S^*|}$, then $\mathbf{x}$ is not optimal for* LP-DenLCS.

Proof. According to Lemma 13, S^* is also the maximum LDS, i.e., no other LDS has cardinality larger than S^*. Prove by contradiction. Assume there is a feasible, optimal solution $\mathbf{x}' = \{x'_v | v \in V\}$ s.t. $\max\{x'_v | v \in V\} < \frac{1}{|S^*|}$ with critical integer k' (defined in Definition 13).

If $k' > |S^*|$, by Lemma 32 we can find a simple feasible solution $\mathbf{x}^+ = \{x_v^+ | v \in V\}$, where $x_v^+ = \frac{\|\mathbf{x}'\|}{k'}$ if $v \in \mathsf{Pre}(\mathbf{x}, k')$ and $x_v^+ = 0$ otherwise. With $f_{\Omega,R}(\mathbf{x}^+) \geq f_{\Omega,R}(\mathbf{x}')$ and Lemma 29, $f_{\Omega,R}(\mathbf{x}^+) = \rho_{\Omega,R}(\mathsf{Pre}(\mathbf{x}, k'))= \rho^*_{\Omega,R}$, contradicting to S^* being the maximum LDS.

If $k' \leq |S^*|$, let $\mathbf{x}^1, \mathbf{x}^2, \cdots, \mathbf{x}^n$ be the peeling of $\mathbf{x}'$ defined by Definition 13, we discuss in two cases, i) $\|\mathbf{x}^i\| = 0$ for all $i \in [k'+1, n]$, and ii) there exists $l > k'$ such that $\|\mathbf{x}^l\| > 0$. We show that in any of the two cases, there will be contradictions and thus complete the proof.

Case(i). From Equation (16), $x_{v_i} = \sum_{j \in [i,n]} \Delta(\mathbf{x}, j)$. By Definition 13, $x_{v_i}^j = \Delta(\mathbf{x}, j)$ if $i \leq j$, so $x_{v_1}^j = \Delta(\mathbf{x}, j)$ and $x_{v_i} = \sum_{j \in [i,n]} x_{v_1}^j$. Note that for Case (i), $x_{v_i}^j = 0$ for $\forall i \in [1, n]$ and $\forall j \in [k'+1, n]$, so $x_{v_i} = 0$ for all $i \in [k'+1, n]$, so $|\{v \in V | x'_v > 0\}| \leq k'$, then because $\max\{x'_v | v \in V\} < \frac{1}{|S^*|}$,

$\|\mathbf{x}'\| < k'\frac{1}{|S^*|} \leq 1$. We then can construct a feasible solution $\mathbf{x}'' = \{x''_v | v \in V\}$ where $x''_v = \frac{x'_v}{\|\mathbf{x}'\|}$ for each $v \in V$. Given $\mathbf{x}''$, define $\mathbf{y}'' = \{y''_e | e \in E\}$ following Lemma 26 such that $y''_e = \min\{x''_u, x''_v\}$ if $\lambda_{|e \cap R|} > 0$, $y''_e = 0$ if $\lambda_{|e\cap R|} = 0$, and $|x''_u - x''_v|$ if $\lambda_{|e\cap R|} < 0$. Then for all $e \in E$, $y''_e = \frac{y'_e}{\|\mathbf{x}'\|}$. Besides, $f_{\Omega,R}(\mathbf{x}'') = \sum_{e\in E} \lambda_{|e\cap R|} y''_e = \sum_{e\in E} \lambda_{|e\cap R|} \frac{y'_e}{\|\mathbf{x}'\|} = \frac{f_{\Omega,R}(\mathbf{x}')}{\|\mathbf{x}'\|}$. Since $f_{\Omega,R}(\mathbf{x}') = \rho_{\Omega,R}(S^*) > 0$, $f_{\Omega,R}(\mathbf{x}'') = \frac{f_{\Omega,R}(\mathbf{x}')}{\|\mathbf{x}'\|} > f_{\Omega,R}(\mathbf{x}')$, contradicting to $\mathbf{x}'$ being an optimal solution.

Case (ii) By Lemma 31 (3), $f_{\Omega,R}(\mathbf{x}') = \sum_{i\in[n]} f_{\Omega,R}(\mathbf{x}^i)$. Further, by Lemma 29, $\sum_{i\in[n]} f_{\Omega,R}(\mathbf{x}^i) = \sum_{i\in[n]} \|\mathbf{x}^i\| \rho_{\Omega,R}(\mathsf{Pre}(\mathbf{x}', i))$. By the setting of Case (ii), k' is the critical integer and $l > k'$, so $\rho_{\Omega,R}(\mathsf{Pre}(\mathbf{x}', l)) < \rho^*_{\Omega,R} = \rho_{\Omega,R}(\mathsf{Pre}(\mathbf{x}', k))$. Recall the setting of Case (ii), s.t., $\|\mathbf{x}^l\| > 0$, it follows that $\|\mathbf{x}^l\|\rho_{\Omega,R}(\mathsf{Pre}(\mathbf{x}', l)) < \|\mathbf{x}^l\|\rho_{\Omega,R}(\mathsf{Pre}(\mathbf{x}', k))$. Since $\|\mathbf{x}^i\|\rho_{\Omega,R}(\mathsf{Pre}(\mathbf{x}', i)) \leq \|\mathbf{x}^i\|\rho_{\Omega,R}(\mathsf{Pre}(\mathbf{x}', k))$ for all $i \in [n]$, $f_{\Omega,R}(\mathbf{x}') = \sum_{i\in[n]} \|\mathbf{x}^i\|\rho_{\Omega,R}(\mathsf{Pre}(\mathbf{x}', i)) < \sum_{i\in[n]} \|\mathbf{x}^i\|\rho_{\Omega,R}(\mathsf{Pre}(\mathbf{x}', k)) = \|\mathbf{x}'\|\rho_{\Omega,R}(\mathsf{Pre}(\mathbf{x}', k)) = f_{\Omega,R}(\mathbf{x}')$, contradiction. Note that the second last equation is supported by Lemma 31(2). Thus, combining Case (i) and Case (ii) under $k' \leq |S^*|$ and the case where $k' > |S^*|$, there does not exist a feasible optimal solution $\mathbf{x}' = \{x'_v | v \in V\}$ s.t. $\max\{x'_v | v \in V\} < \frac{1}{|S^*|}$. □

A.5 Proof of Lemma 20

As S_i is an LDS on L_i, $\rho_{\Omega,R}(S_{i+1} \cap V_i | L_i) \leq \rho_{\Omega,R}(S_i|L_i)$. By Lemma 11, if $S_{i+1} \setminus V_i \neq \emptyset$, then $\rho_{\Omega,R}(S_{i+1}|L_{i+1}) < \rho_{\Omega,R}(S_{i+1} \cap V_i|L_i) \leq \rho_{\Omega,R}(S_i|L_i)$; else, $\rho_{\Omega,R}(S_{i+1}|L_{i+1}) \leq \rho_{\Omega,R}(S_{i+1} \cap V_i|L_i) \leq \rho_{\Omega,R}(S_i|L_i)$. It follows that when $\rho_{\Omega,R}(S_i|L_i) = \rho_{\Omega,R}(S_{i+1}|L_{i+1})$, $S_{i+1} \subseteq V_i$ and $\mathcal{N}^+(S_{i+1}) \subseteq V_{i+1}$. If $S_{i+1} \setminus V(L_i) \neq \emptyset$, then $\rho_{\Omega,R}(S_i|L_i) > \rho_{\Omega,R}(S_{i+1}|L_{i+1})$, contradiction. Therefore, $S_{i+1} \subseteq V_i$. Denote by C_{i+1} the core set at iteration $i+1$. If $S_{i+1} \subseteq S_i$, since at iteration $i+1$, $S_i \subseteq C_{i+1}$, thus $\mathcal{N}^+(S_i) \subseteq V_{i+1}$, and thus $\mathcal{N}^+(S_{i+1}) \subseteq V_{i+1}$ which terminates the loop. Otherwise, $S_{i+1} \setminus S_i \neq \emptyset$, which we show below that it will lead to a contradiction. By Lemma 11 and $S_{i+1} \subseteq V_i$, $\rho_{\Omega,R}(S_{i+1}|L_i) \geq \rho_{\Omega,R}(S_{i+1}|L_{i+1})$. Since $[\rho^*_{\Omega,R}|L_i] = \rho_{\Omega,R}(S_i|L_i) = \rho_{\Omega,R}(S_{i+1}|L_{i+1})$, it follows that $\rho_{\Omega,R}(S_{i+1}|L_i) \geq [\rho^*_{\Omega,R}|L_i]$, i.e., S_{i+1} is also optimal in L_i. Therefore, by Lemma 13, $\rho_{\Omega,R}(S_i \cup S_{i+1}|L_i) = [\rho^*_{\Omega,R}|L_i]$ where $S_i \cup S_{i+1} \setminus S_i \neq \emptyset$, contradicting the fact that $\mathcal{A}$ returns a maximal LDS (Lemma 8) of L_i .

A.6 Proof of Lemma 21

In ExpansionFramework, consider iteration l with $0 \leq l < k$, the working subgraph is $L_l(V_l, E_l)$ and LDS is $S_l \subseteq V_l$. By Lemma 14, $\rho_{\Omega,R}(S_l|L_l) \geq 1$ and thus $2 \leq |S_l| \leq g_{\Omega,R}(S_l|L_l)$. Lemma 15: $g_{\Omega,R}(S_l|L_l) \leq 2\mathrm{vol}(R) \doteq U_s$, thus $|S_l| \leq U_s$.

If the weights in Ω are integers, then $g_{\Omega,R}(S_l)$ is an integer. As the density value strictly decreases in each iteration before termination according to Lemmas 20, iteration l needs to have a different value of $\rho_{\Omega,R}(S_l|L_l)$ from all previous iterations, so the pair of $(g_{\Omega,R}(S_l), |S_l|)$ has $U_s(U_s-1)$ possible values. So $k-1 \leq U_s(U_s-1)$. As $\mathrm{vol}(R) \geq 1$, $k \leq U_s^2 = O(\mathrm{vol}^2(R))$.

Recall that in the input tuple $\Omega \in C_L$. According to Lemma 12, $\Omega(p) \neq 0$ only if $p \in I_L = \{(1,1),(1,2),(2,3),(2,4)\}$, in other words, $g_{\Omega,R}(S_l) = \sum_{p\in I_L} \Omega(p)|E^l_p|$ where $\{E^l_p | p \in \mathcal{I}\}$ denotes the edge partitioning $\mathcal{E}(S_l, R|L_l)$. Note that $S_l = V_1 \cup V_2$ where node partitioning $V_1 \doteq S_l \cap R$ and $V_2 \doteq S_l \setminus R$. Observe that for any $p \in \mathcal{I}_L$, $p \cap \{1,2\} \neq \emptyset$, in other words, $E^l_p \subseteq ((V_1 \cup V_2) \times V) \cap E = E^+(V_1 \cup V_2) = E^+(S_l)$. Thus, $0 \leq |E^l_p| \leq |E^+(S_l)|$ where $|E^l_p|, |E^+(S_l)| \in \mathbb{Z}$. Also, since $|S_l|$ is bounded by U_s by Lemma 15 and $\deg(v)$ for each $v \in S_l$ is bounded by U_d by Lemma 18, $|E^+(S_l)| \leq U_s U_d$, i.e, bounded by their multiplication. Since each iteration l has a different value of $\rho_{\Omega,R}(S_l|L_l)$, tuple $(|E^l_{11}|, |E^l_{12}|, |E^l_{23}|, |E^l_{24}|, |S_l|)\} \in \mathbb{Z}^5$ will also be different for each iteration l. Recall that $0 \leq |E^l_p| \leq U_s U_d$ for $p \in I_L$ and $2 \leq |S_l| \leq U_s$, $k \leq (U_s U_d + 1)^4 (U_s - 1) + 1 \leq (U_s U_d + 1)^4 U_s = O(\mathrm{vol}^9(R))$.

Received December 2023; revised February 2024; accepted March 2024