

# $k$ -コーダルグラフにおける最適な低競合ショートカット

北川 大貴<sup>1</sup> 北村 直暉<sup>1</sup> 大館 陽太<sup>2,a)</sup> 泉 泰介<sup>1,b)</sup>

概要：本稿は、*CONGEST* モデル上の分散アルゴリズム設計における重要な基本手法の一つである、低競合ショートカットを取り扱う。特に、入力グラフを  $k$ -コーダルグラフとした場合、良い低競合ショートカットが存在し、かつ 1 ラウンドでそれを (分散的に) 構成できることを示す。具体的には、任意の  $n$  頂点  $k$ -コーダルグラフに対して、クオリティ  $O(kD)$  の低競合ショートカットの構成アルゴリズムを提案する ( $D$  はグラフの直径)。この結果は、 $k$  があまり大きくない場合、 $k$ -コーダルグラフは一般のグラフにおける低競合ショートカットのクオリティの下界  $\Omega(\sqrt{n} + D)$  よりも大幅に良いショートカットを持つことを意味している。また、 $k \leq D$  かつ  $kD \leq \sqrt{n}$  の条件のもとで、提案アルゴリズムが最適であること、すなわち低競合ショートカットのクオリティが  $\Omega(kD)$  となるインスタンスが存在することも示す。

キーワード：分散システム, *CONGEST* モデル, 低競合ショートカット,  $k$ -コーダルグラフ

## Optimal Low-Congestion Shortcuts for $k$ -Chordal Graphs

HIROTAKA KITAGAWA<sup>1</sup> NAOKI KITAMURA<sup>1</sup> YOTA OTACHI<sup>2,a)</sup> TAISUKE IZUMI<sup>1,b)</sup>

### 1. はじめに

#### 1.1 研究背景

分散グラフアルゴリズムとは、複数の計算機を相互接続したネットワークにおいて、ネットワークそれ自体をグラフアルゴリズムの入力とみなして、その上で各種問題 (最短経路, 最小生成木, 最小カット等) を解く枠組みである。分散グラフアルゴリズムにおける代表的なモデルのひとつとして、*CONGEST* モデルが存在する。*CONGEST* モデルにおいては、各計算ノードは同期ラウンドに従って動作し、1 ラウンドあたり各通信リンクは  $O(\log n)$  ビットのメッセージを送信可能である ( $n$  は全ノード数) ことを仮定したモデルである。*CONGEST* モデルにおいては通信リンクの帯域が制限されているため、ある 1 台の計算機がネットワーク全体のトポロジ情報を収集して、その上で逐次アルゴリズムを実行するというアプローチでは自明に  $\Omega(n)$

ラウンドの実行時間を必要とする。そのため、*CONGEST* モデルにおけるアルゴリズム設計の興味は、 $n$  に対して準線形時間のアルゴリズムを構成できるかどうかという点にある。

本稿では *CONGEST* モデルにおける問題のひとつである同時部分最小値問題を扱う。グラフの頂点が各々  $O(\log n)$  ビットの値を持つものとし、頂点集合の互いに疎な連結誘導部分グラフへの分割が与えられているとする。同時部分最小値問題は、それぞれの部分グラフ内において、ノード保有値の最小値を求めるといった問題である。同時部分最小値問題は分散システムにおいて最小全域木の構築、最小カット近似など様々な問題への応用がある [1]。その中でも最小全域木の構築とは強く関係しており、頂点数  $n$  のグラフに対して同時部分最小値問題を  $O(f)$  時間で解くアルゴリズムが存在するとき、 $\tilde{O}(f)$  時間<sup>\*1</sup> で最小全域木の構成が可能である [1], [2]。またこのことから逆に、最小全域木の下界が与えられると同時部分最小値問題の下界を求めることも可能である。Peleg と Rubinfeld らは、直径が

<sup>1</sup> 名古屋工業大学大学院工学研究科。466-8555 名古屋市昭和区御器所町。

<sup>2</sup> 熊本大学大学院先端科学研究部。860-8555 熊本市中央区黒髪 2 丁目 39 番 1 号。

a) otachi@cs.kumamoto-u.ac.jp

b) t-izumi@nitech.ac.jp

\*1  $\tilde{O}(\cdot)$  は、通常の  $O(\cdot)$  記法から、 $\text{polylog}(n)$  の項を (相対的に十分小さい項として) 無視した記法である。

$D$ , 頂点数  $n$  のグラフについて最小全域木の構築にかかる時間の下界が  $\Omega(\sqrt{n} + D)$  ラウンドであることを証明している [3], [4], [5]. また, 一般のグラフに対して  $\tilde{O}(\sqrt{n} + D)$  時間で同時部分最小値問題を解くアルゴリズムが存在することも知られている [6]. したがって対数領域の時間複雑性を無視すると, 一般のグラフについては同時部分最小値問題を最適な時間で解くアルゴリズムが知られていることになる. ただし,  $\Omega(\sqrt{n} + D)$  ラウンドの下界に言及するときに注意すべき点として,  $\Omega(\sqrt{n})$  ラウンドの下界はある種の特異なインスタンスに対して成立するラウンド数下界であることに対して,  $\Omega(D)$  ラウンドの下界はインスタンスによらず常に成立する下界 (ユニバーサル下界) であることを指摘しておきたい (任意のグラフにおいて, グラフ全体を一つの部分グラフとした場合の同時部分最小値問題は自明に  $\Omega(D)$  ラウンドを必要とする). そのため, 一般にあるグラフ  $G$  が与えられたとき, その上での最適な同時部分最小値問題の求解ラウンド数に対する漸近的評価は, 必ず  $\tilde{O}(\sqrt{n} + D)$  と  $\Omega(D)$  の間に存在することがわかる. このことは, 同問題に対して, 制限されたグラフクラスに対する計算量解析を検討するための動機を与えることになる.

上述の背景をに基づき, 本稿ではグラフクラスをコーダルグラフの一般化である  $k$ -コーダルグラフに制限した場合の, 同時部分最小値問題の高速な解法について検討する. 特に, 解法へのアプローチとして, 低競合ショートカットと呼ばれる枠組みに注目する. 低競合ショートカットは, 同時部分最小値問題を高速に解くための手法として Ghaffari と Haeupler により提案され [1], 近年多くの注目を集めている. 詳細な定義は後に述べるが, 一般に成立する事実として, クオリティと呼ばれるパラメタが  $q$  であるような低競合ショートカットを  $O(f)$  時間で構成可能である場合, 同時部分最小値問題を  $\tilde{O}(f + q)$  時間で解くことが可能であることが知られている [1]. よって本研究では,  $k$ -コーダルグラフに対する良いクオリティの低競合ショートカットの存在性および構成可能性について論じていく. 提案する具体的な結果は以下の 2 つの事実である.

- 任意の  $n$  頂点  $k$ -コーダルグラフに対して, クオリティ  $O(kD)$  の低競合ショートカットを構成する実行時間 1 ラウンドのアルゴリズムが存在する. この結果は,  $k = O(1)$  であるとき,  $\Omega(D)$  ラウンドのユニバーサル下界に一致する. すなわち, 小さい  $k$  に対して  $k$ -コーダルグラフ上では同時部分最小値問題を一般のグラフに比べて大幅に高速に解くことが可能であることを示している.
- $k \leq D$  かつ  $kD \leq \sqrt{n}$  の条件のもとで, 低競合ショートカットのクオリティが  $\Omega(kD)$  となるようなインスタンスが存在する. この結果は提案するアルゴリズムが (ショートカットの構成可能性の意味において) 最適であることを示している.

## 1.2 関連研究

最小生成木問題は分散グラフアルゴリズムの設計におけるもっとも重要な基本問題の一つであり, それ単体が重要であるだけでなく, 他の数多くのアルゴリズム設計のツールとしての役割も持つ. そのため, 最小生成木構成のための効率のよいアルゴリズムの設計は, 分散グラフアルゴリズム設計の黎明期より数多くの研究が示されており [2], [6], [7], また, その計算量下界についても研究が進んでいる [3], [5].

前述の通り, 最小生成木問題の高速な求解のためのキーポイントは, 同時部分最小値問題に対する高速なアルゴリズムの設計にある. 低競合ショートカットは同問題を高速に解くための枠組みとして提案された. 現時点で, 平面グラフ及び種数制限グラフ [1], [8], 部分  $K$  木 [8], 禁止マイナー族 [9] により特徴づけられるグラフ等に対して, 一般のグラフにおける  $\Omega(\sqrt{n} + D)$  の下界を破るクオリティを持つ低競合ショートカットの存在が示されている. また, 平面グラフ (クオリティ  $\tilde{O}(D)$ ), 種数  $g$  のグラフ (クオリティ  $\tilde{O}(\sqrt{g}D)$ ), 部分  $k$  木 (クオリティ  $\tilde{O}(kD)$ ) については, 現在知られている低競合ショートカットの構成法がクオリティの意味で最適であることも示されている [1], [8]. 低競合ショートカットのアルゴリズム設計への応用としては, 最小生成木および最小カット近似 [1], 単一始点最短経路問題 [10], 平面グラフに対する深さ優先探索木の構成 [11] 等が知られている.

## 1.3 論文の構成

本稿は全 4 節で構成される. 2 節では, 本稿で考える CONGEST モデルと同時部分最小値問題, ならびに低競合ショートカットの定義を述べる. 3 節では, 本稿の主定理およびその証明をおこなう. 4 節では, まとめと今後の課題について述べる.

## 2. 諸定義

### 2.1 モデル定義

本稿で考える分散システム (CONGEST モデル) は, 単純無向連結グラフ  $G = (V, E)$  により表現される. ここで  $V$  はノードの集合であり,  $E$  は通信リンクの集合である. また,  $|V| = n$  とする. CONGEST モデルでは計算機はラウンドに従って同期して動作するものとする. 1 ラウンド内で, 隣接頂点へのメッセージ送信, 隣接頂点からのメッセージ受信, 内部計算 (多項式時間) をおこなう. 各辺は単位ラウンドあたり  $O(\log n)$  ビットを双方向に伝送可能であり, 各ノードは異なる接続辺に異なる内容のメッセージを同一ラウンドに送信可能である. 各ノードはまた,  $O(\log n)$  ビットの自然数値による ID が付与されており, 自身の隣接頂点全ての ID を既知であるとする. 各ノードはグラフのトポロジに関する事前知識を持たないものとする.

る。ただし、本稿では一般のグラフではなく、制限されたグラフクラス ( $k$ -コーダルグラフ) を考えており、アルゴリズムはトポロジ  $G$  が当該のクラスに属していることを仮定して動作する。

## 2.2 $k$ -コーダルグラフ

コーダルグラフの一般化である  $k$ -コーダルグラフは次のように定義される。

**定義 2.1.** あるグラフ  $G$  について、 $G$  中の長さ  $k+1$  以上の任意のサイクルが必ず弦を持つとき、そのグラフは  $k$ -コーダルであると呼ぶ。

特に  $k=3$  の場合は単にコーダルグラフと呼ばれ、区間グラフ等に代表される、各種交差グラフ族との関連が深いクラスであることが知られている。コーダルグラフは明らかに平面的でないグラフを含み、また種数および木幅に関する非自明な上界を持たない。さらに、任意の大きさのクリークを含み得るため、いかなる禁止マイナーも持たない。そのため、すでに良い低競合ショートカットの存在が知られているいかなるグラフクラスにも包含されないため、既知の手法を用いて効率的な低競合ショートカットを構成することはできない。

## 2.3 同時部分最小値問題

同時部分最小値問題は、*CONGEST* モデルにおける基本問題の一つであり、以下のように定義される。

- 入力:各頂点  $v \in V$  に対する  $O(\log n)$  ビットの入力値  $x_v \in \mathbb{N}$ 、ならびにグラフ  $G$  の互いに素な連結部分グラフへの分割  $P_1, P_2, \dots, P_N$  (すなわち、各  $P_i$  は  $G$  中の連結な部分グラフであり、 $V(P_i)$  の和は  $V(G)$  に一致する)。
- 出力:各  $P_i (1 \leq i \leq N)$  中のノードは、 $V(P_i)$  中のノードの入力値の最小値  $\min_{v \in V(P_i)} x_v$  を出力する。

上述の入力における各部分グラフ  $P_i$  を、以降パートと呼ぶ。同時部分最小値問題のナイーブな解法は、各パート  $P_i$  中において独立に最小値問題を解くことである。具体的には、まず  $P_i$  に対して根付きの幅優先木 (根はリーダ選挙を用いて決定する) を構成して、各ノードは根に向かって自分がこれまでに受け取った値 (自身の値と、子より転送されてきた値) のうち最小のものを転送していく。木の高さに比例する時間で根は最小値を求めることができるので、その後根は求めた最小値を木上でブロードキャストすることで最小値発見問題を解くことができる。このアルゴリズムの計算時間は構成する木の高さに比例するが、一般に  $G$  の誘導部分グラフの直径は  $G$  の直径とは無関係に大きくなりえる (最悪時は  $\Omega(n)$ )。そのため、このアプローチで達成可能な計算時間の上界は  $O(n)$  ラウンドとなる。別の手法としては、グラフ  $G$  全体で幅優先木  $T$  を構成して、全てのパートに対して  $T$  を用いて最小値発見を行うという方

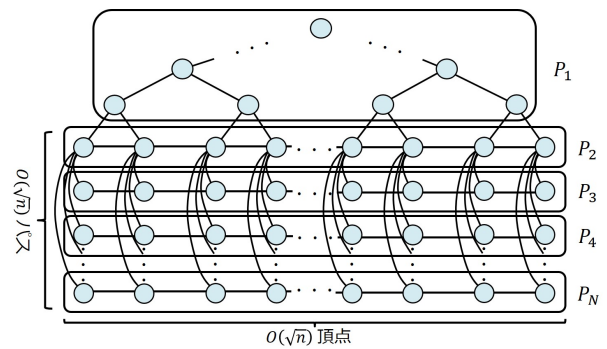


図 1  $\tilde{\Omega}(\sqrt{n})$  ラウンドを必要とするインスタンス

法が考えられる。この方法は  $T$  の高さを  $O(D)$  で押さえることが可能であるが、一方で最小値発見を行うグループは  $N$  個存在するため、 $T$  上で  $N$  回最小値発見問題を解く必要がある。パイプラインスケジューリングを用いることでその計算時間を  $O(D+N)$  時間に抑えることは可能であるが、 $N = \Omega(n)$  となりえるため、この手法においても達成可能な計算時間の上界はやはり  $O(n)$  ラウンドとなる。

一般のグラフにおける最適な解法はこれら二つを折衷することで達成される。具体的には、各パートごとに、パート内の頂点数が  $\sqrt{n}$  以下であるならば前者の手法を、パートの直径が  $\sqrt{n}$  以上であるならば後者の手法を用いる。このとき、 $(\sqrt{n} + D)$  ラウンドで同時部分最小値問題を解くことが可能である。なお、いかなるアルゴリズムを用いても同時部分最小値問題を解くためには  $\tilde{\Omega}(\sqrt{n})$  ラウンドを必要とするようなインスタンスの存在 (図 1) が知られており、上述の手法は一般のグラフにおいて最適なアルゴリズムである。

**定理 2.2** ([1], [3]). 任意の  $n$  頂点グラフ  $G$  およびパートへの分割  $P_1, P_2, \dots, P_N$  に対して、 $\tilde{O}(\sqrt{n} + D)$  ラウンドで同時部分最小値問題を解くアルゴリズムが存在する。また、同時部分最小値問題を解くいかなるアルゴリズムについても、その実行時間が  $\tilde{\Omega}(\sqrt{n})$  ラウンドとなるような直径  $O(\log n)$  のグラフおよびパートへの分割が存在する。

なお、この下界は、アルゴリズムとして乱択を許した場合にも成立する。

## 2.4 $(d, c)$ -低競合ショートカット

$O(\sqrt{n} + D)$  ラウンドで同時部分最小値問題を解くアルゴリズムのキーアイデアは、各パートで最小値の発見に用いる木に対して、その高さ (dilation) と利用回数 (congestion) を  $\Theta(\sqrt{n})$  でバランスさせた点にある。低競合ショートカットは、この考え方を一般化したものとみなすことができる。以下、形式的な定義を述べる。

**定義 2.3** ([1]). グラフ  $G = (V, E)$  と  $G$  の互いに素な連結誘導部分グラフ  $P_1, P_2, \dots, P_N$  への分割が与えられているとする。  $(d, c)$ -低競合ショートカットは各  $P_i$  に対して以下のように定義される部分グラフ  $H_1, H_2, \dots, H_N \subset G$  の集

合である。

(1) 各  $i$  において,  $G$  の部分グラフ  $P_i + H_i$  の直径は高々  $d$  である. (low dilation)

(2) 各辺  $e \in E$  において, 部分グラフ  $P_i + H_i$  に含まれる回数は高々  $c$  回である. (low congestion)

一般に,  $(d, c)$ -低競合ショートカットの構築にかかる時間を  $f$  とすると,  $\tilde{O}(d + c + f)$  時間で同時部分最小値問題を解くアルゴリズムが知られている [1], [12]. そのため, 低競合ショートカットの良さを評価する指標として,  $d, c$  を別々に検討する必要はなく, その最大値のみで十分である. 以降,  $(d, c)$ -低競合ショートカットに対して決まる値  $\max\{d, c\}$  を  $(d, c)$ -低競合ショートカットのクオリティと呼ぶ.

### 3. 本研究の主定理と証明

本研究の主結果の一つは, 任意の  $k$ -コーダグラフは良いクオリティの低競合ショートカットを持ち, かつそれを (分散的に) 構成可能であるということである.

**定理 3.1.** 任意の  $k$ -コーダグラフに対して, そのクオリティが  $O(kD)$  となる低競合ショートカットが存在し, またそれを 1 ラウンドで構成する CONGEST モデル上の分散アルゴリズムが存在する.

この定理は, 一般に  $k = \tilde{O}(1)$  であるならば, ほぼ最適に近いショートカットが存在することを示している. 一方で, ショートカットのクオリティと  $k$  の値との依存性に関して, この結果がほぼ最適であることも示すことができる. 具体的には, 以下のような定理を得る.

**定理 3.2.**  $k \leq D, kD \leq \sqrt{n}$  であるような任意の  $k$  および  $D$  に対して, 任意のショートカットのクオリティが  $\Omega(kD)$  となるような,  $n$  頂点, 直径  $D$  の  $k$ -コーダグラフ  $G$  とその上の分割  $P_1, P_2, \dots, P_N$  が存在する.

本節では以下, この 2 つの定理の証明を行っていく.

#### 3.1 定理 3.1 の証明

まずは定理 3.1 の証明をおこなう. 定理 3.1 は, 具体的にショートカットを構成するアルゴリズムを示すことにより示す. アルゴリズムは以下に示すとおりであり, 極めて単純である.

任意の分割  $P_i \subseteq V$  について, そこに属するすべての頂点  $v \in P_i$  は, 自身に接続している辺すべてを,  $P_i$  のショートカット辺として  $H_i$  に追加する.  $v$  は各接続辺  $(v, u)$  について, それを  $H_i$  に加えたことを隣接頂点  $u$  に通知する.

このアルゴリズムは, 明らかに 1 ラウンドで実行可能である. また, 分割は互いに素であるため, 各辺は高々 2 つのグループに属する. よって定理 3.1 を示すためには, 任意の  $i \in [1, N]$  について  $P_i + H_i$  の直径が  $O(kD)$  となることを示せば十分である. すなわち, 証明のゴールは以下の補

題を示すことである.

**補題 3.3.** 任意の  $i \in [1, N]$  に対して,  $G_i = H_i + P_i$  とすると,  $G_i$  の直径は高々  $kD$  である.

まず, 証明で使用する記号の定義をおこなう.  $S_{uv} = (s_0, s_1, \dots, s_l)$  を  $G$  における  $u, v$  間の最短パスを成す頂点系列とする. また  $S'_{uv} = (s'_0, s'_1, \dots, s'_l)$  を  $P_i$  における  $u, v$  間の最短パスを成す頂点系列とする.  $S_{uv}$  と  $S'_{uv}$  のいずれも, 複数存在する場合は任意の 1 つを選ぶものとする. 定義より明らかに  $s_0 = s'_0 = u, s_l = s'_l = v$  となる.  $S_{uv}, S'_{uv}$  は系列であるが, 曖昧さのない範囲で頂点集合としても取り扱うものとする. 共通集合  $S_{uv} \cap S'_{uv}$  について, その要素を  $S_{uv}$  に現れる順番に並べた系列を  $T_{uv} = (t_0, t_1, \dots, t_m)$  とする.  $t_x = s_y$  を満たす  $y$  を  $c(x)$ ,  $t_x = s'_y$  を満たす  $y$  を  $c'(x)$  で表すものとする. また,  $S_{uv}[i, j]$  を  $S_{uv}$  の中の  $s_i$  から  $s_j$  までの部分系列  $(s_i, s_{i+1}, \dots, s_{j-1}, s_j)$  とし,  $S'_{uv}[i, j]$  を  $S'_{uv}$  の中の  $s'_i$  から  $s'_j$  までの部分系列  $(s'_i, s'_{i+1}, \dots, s'_{j-1}, s'_j)$  とする.  $S_{uv}[i, j]$ , および  $S'_{uv}[i, j]$  はいずれも当該部分系列のパスに相当する  $G$  の部分グラフの意味合いでも使う場合がある. なお,  $i < j$  であることは必ずしも要求しない.  $i > j$  の場合は部分列は元のパスを逆向きにたどる系列に相当する. これらの記法をまとめたものを図 2 に示す.

補題 3.3 の証明のために, まず以下の補題を示す.

**補題 3.4.** 任意の  $x (1 \leq x \leq m - 1)$  について,  $\text{dist}_{G_i}(t_x, t_{x+1}) \leq k(c(x+1) - c(x))$

*Proof.*  $G' = S'_{uv}[c'(x), c'(x+1)] + H_i$  とする. 任意の  $j (0 \leq j < |c(x+1) - c(x)|)$  について, ある  $u_j \in S_{uv}[c(x) + j, c(x+1)]$  が存在して,  $\text{dist}_{G'}(s_{c(x)}, u_j) \leq kj$  が成立することを,  $j$  に関する帰納法により示す ( $j = |c(x+1) - c(x)|$  とすることで題意が証明される). なお, 系列  $S'_{uv}[c'(x), c'(x+1)]$  に関しては,  $c(x) > c(x+1)$  の可能性があり得るが, 以降は  $c(x) < c(x+1)$  であることを仮定して証明する ( $c(x) > c(x+1)$  のケースは同様に証明可能). また, 帰納段階の記号等をまとめたものを図 3 に示す. (基底段階  $j = 0$  のときは明らかである. (帰納段階) ある  $j$  までの成立を仮定する. このとき  $u_j = s_{c(x)+j}$  とする.  $\hat{j} > j+1$  のときは  $u_{j+1} = \hat{j}$  が明らかに補題 3.4 を満たすので,  $\hat{j} = j+1$  のときのみを考えれば良い.  $e = (u_j, s'_{c'(x)+h}) \in H_i$  とする. このとき  $e$  の候補が複数あるときは  $h$  が最大のものを  $e$  とする.  $S_{uv}[c(x) + \hat{j}, c(x+1)], e, S'_{uv}[c'(x) + h, c'(x+1)]$  で構成されるサイクル  $C$  を考えると, サイクルの大きさが  $k+1$  以上のとき, グラフ  $G$  が  $k$ -コーダであることより必ず弦をもつ. そのような弦を  $e' = (s_{c(x)+y}, s'_{c'(x)+y'}) \in H_i$  とする (複数個本ある場合は  $y'$  が最も小さいものを選ぶ). このとき辺  $e$  として  $u_j$  に接続する辺のうち  $h$  の値が最大となるものを取っているため,  $e'$  は  $u_j$  には接続していない. すなわち,  $y > \hat{j}, y' \geq h$  である. パス  $S'_{uv}[c'(x) + h, c'(x) + y']$  の長さは  $k$ -コーダグラフの性質より  $k-3$  以下になる. よっ

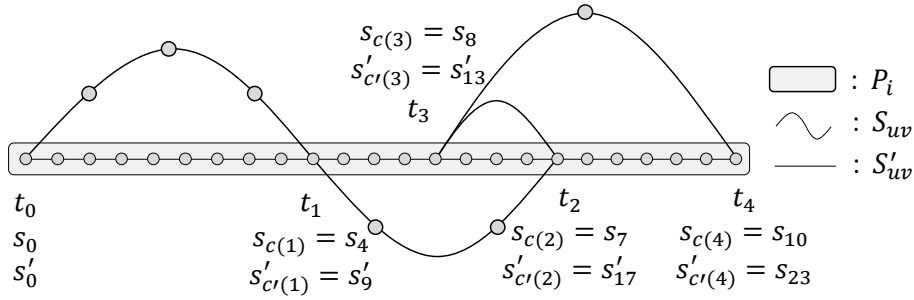


図 2 記号の定義とその例

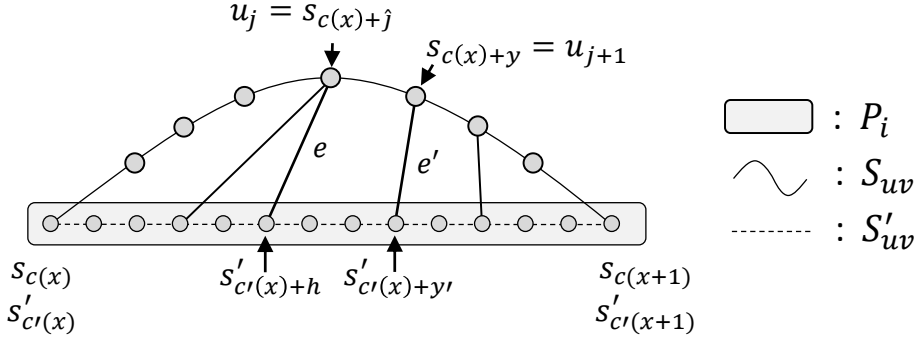


図 3 補題 3.4 の証明

て  $u_j$  から  $s_{c(x)+y}$  へのパス  $S'_{uv}[c'(x)+h, c'(x)+y'] + \{e, e'\}$  は長さ  $k-1$  以下であり,  $\text{dist}_{G'}(u_j, s_{c(x)+y}) \leq k$  を得る. 帰納法の仮定より  $\text{dist}_{G'}(s_{c(x)}, u_j) \leq kj$ .  $u_{j+1} = s_{c(x)+y}$  であるので  $\text{dist}_{G'}(s_{c(x)}, s_{c(x)+y}) \leq k(j+1)$  が成り立ち,  $u_{j+1} = s_{c(x)+y}$  とすることで,  $j+1$  についても命題が成立する. 以上により示された.  $\square$

上記補題を用いることで, 補題 3.3 を示す.

*Proof.* 補題 3.4 の式を全ての  $x$  について足し合わせることで, 以下の式を得る.

$$\begin{aligned} \text{dist}_{G_i}(t_0, t_m) &\leq \sum_{0 \leq i \leq m-1} \text{dist}_{G_i}(t_i, t_{i+1}) \\ &\leq \sum_{0 \leq i \leq m-1} k(c(i) - c(i+1)) \\ &\leq k(c(m) - c(0)) \\ &\leq kD \end{aligned}$$

$\square$

### 3.2 定理 3.2 の証明

次に定理 3.2 の証明をおこなう. まず, 証明に先立ち, 下界を与えるような  $G$  の構成を述べる. グラフ  $G$  の構成は  $k$  以外に 2 つのパラメタ  $x, N$  を持つため, 以降  $G(k, x, N)$  として記載する. パラメタ  $x$  および  $N$  は最終的に定理 3.2 の条件を満たすように調整される. 簡便のため  $K = k/2 - 1$  と定める.  $G(k, x, N)$  の頂点集合および辺集合をそれぞれ  $V(k, x, N)$  および  $E(k, x, N)$  で表すとしたとき, それらは以下のように定義される.

$$\begin{aligned} V(k, x, N) &= \{v_{1,j} \mid 0 \leq j \leq x\} \\ &\cup \{v_{i,j} \mid 2 \leq i \leq N, 0 \leq j \leq xK\}. \end{aligned}$$

$$E(k, x, N) = E_1 \cup E_2 \cup E_3 \cup E_4 \quad \text{s.t.}$$

$$\begin{aligned} E_1 &= \{\{v_{1,j}, v_{1,j+1}\} \mid 0 \leq j \leq x-1\}, \\ E_2 &= \{\{v_{i,j}, v_{i,j+1}\} \mid 2 \leq i \leq N, 0 \leq j \leq xK-1\}, \\ E_3 &= \{\{v_{1,j}, v_{i,h}\} \mid 2 \leq i \leq N, 0 \leq j \leq x, h = jK\}, \\ E_4 &= \{\{v_{i,h}, v_{j,h}\} \mid 2 \leq i, j \leq N, i \neq j, h \bmod K = 0\}. \end{aligned}$$

また, 各パートへの分割  $\mathcal{P} = \{P_1, P_2, \dots, P_N\}$  は以下のように定義される.

$$P_i = \begin{cases} \{v_{1,j} \mid 0 \leq j \leq x\}, & i = 1 \\ \{v_{i,j} \mid 2 \leq i \leq N, 0 \leq j \leq xK\}, & i > 1. \end{cases}$$

グラフ  $G(k, x, N)$  およびパート集合  $P_1, P_2, \dots, P_N$  の構成を図 4 に示す.

下界の証明に先立って, まずはグラフ  $(k, x, N)$  が  $k$ -コーダルであることを証明する.

**補題 3.5.** グラフ  $G(k, x, N)$  に含まれる任意の単純サイクルは, 長さ  $k$  以下であるか, もしくは弦をもつ.

*Proof.* 簡単のために, 頂点の一部に以下のような別名  $v'_{xy}$  を与える.

- $v'_{1,j} = v_{1,j} (0 \leq j \leq x)$
- $v'_{i,j} = v_{i,h} (2 \leq i \leq N, 0 \leq j \leq x, h = jK).$

行と列と呼ばれる頂点の部分集合を定義する.  $i$  行目の頂点集合を  $R_i = \{v'_{i,j} \mid 0 \leq j \leq x\}$ ,  $i$  列目の頂点集合を

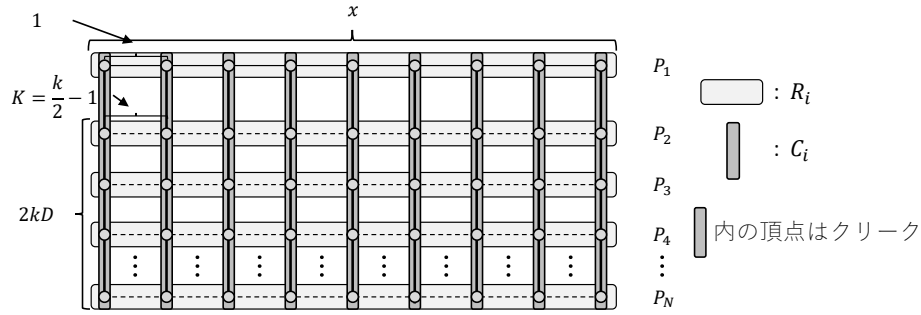


図 4  $k$ -コードグラフ  $G(k, x, N)$

$C_i = \{v'_{j,i} | 1 \leq j \leq N\}$  とする．以下の証明で  $R_i, C_i$  の  $i$  をそれぞれ行のインデックス，列のインデックスと呼ぶ．

今， $G(k, x, N)$  中の任意のサイクル  $X$  を考える． $X$  の頂点集合が交わる列のうち，最小のインデックスを  $l$ ，最大のインデックスを  $r$  とする．また，行についても同様に， $X$  が交わる行のうち最小，最大のインデックスをそれぞれ  $t$  および  $b$  で表す．また， $X$  の頂点集合が交わる列のうち，交わりが最大となるような列のインデックスを  $m$  とし，その交わりに含まれる頂点数を  $a_m$  で表すとす．このとき，任意のサイクル  $X$  は以下の 4 つのケースの少なくとも一つにあてはまる．

- (1)  $r - l \geq 2$  である
- (2)  $a_m \geq 3$  である
- (3)  $r - l = 0$  である
- (4)  $r - l = 1$  であり  $a_m = 2$  である

これらすべてについて補題 3.5 が成り立つことを示す．また，以下の証明を図 5 に示す．

- (1)  $r - l \geq 2$  の場合：グラフ  $G(k, x, N)$  の構成より， $l$  列中の頂点と  $r$  列中の頂点を結ぶ任意のパスはかならず  $l+1$  列と交わりを持つ． $X$  が単純サイクルであることより，このとき  $X$  は  $l+1$  列目と交わりを持ち，またその大きさは少なくとも 2 である． $u, v$  を  $X$  と  $C_{l+1}$  の交わりとすると， $C_{l+1}$  はクリークを成しているのて， $u$  と  $v$  は隣接している．すなわち，辺  $(u, v)$  が  $X$  における弦となる．
- (2)  $a_m \geq 3$  である場合：このとき， $C_m$  中に， $X$  で隣接していない 2 頂点が存在する． $C_m$  がクリークであることよりそれらの 2 頂点間は辺をもち，その辺が  $X$  における弦となる．
- (3)  $r - l = 0$  である場合： $X$  自体がクリークであり，自明．
- (4)  $r - l = 1$  かつ  $a_m = 2$  である場合： $X$  が  $v'_{t,l}, v'_{t,r}, v'_{b,l}, v'_{b,r}$  の 4 頂点と  $v'_{t,l}, v'_{t,r}$  を結ぶパス， $v'_{b,l}, v'_{b,r}$  を結ぶパスで構成されていることがわかる．このとき， $\text{dist}(v'_{t,l}, v'_{t,r}) \leq K = k/2 - 1$ ， $\text{dist}(v'_{b,l}, v'_{b,r}) = K = k/2 - 1$ ， $\text{dist}(v'_{t,l}, v'_{b,l}) = \text{dist}(v'_{t,r}, v'_{b,r}) = 1$  となり， $X$  の長さは  $k$  以下となる．

以上により示された．□

次の補題によりグラフ  $G(k, x, N)$  に対するショートカットのクオリティの下界を与える．

**補題 3.6.**  $D > 2K$ ， $N \geq 2kD$  とすると，グラフ  $G(k, D - K, N)$  の直径は高々  $D$  であり， $\mathcal{P}$  に対する任意のショートカットのクオリティは  $\Omega(kD)$  である．

*Proof.*  $D > 2K$  のとき，任意の 2 頂点間に高々長さ  $D$  のパスが存在することは容易に確認できる．ショートカットのクオリティに関する証明は背理法による．グラフ  $G(k, D - K, N)$  のショートカットのクオリティが  $o(kD)$  であると仮定する．このとき， $P_1$  に含まれる辺のショートカットとしての総使用回数  $U$  は， $U \leq (D - K)kD$  となる．よって各パートがショートカットの辺として  $P_1$  中の辺を使える回数の平均は  $(D - K)kD/2kD = (D - K)/2$  となり，少なくとも 1 つのパート  $P_z$  が高々  $\frac{(D-K)}{2}$  本の辺を利用している．今， $H$  を  $P_z$  とそのショートカット辺からなる部分グラフとする．また， $Z$  を  $P_1$  中の辺で  $P_z$  がショートカット辺として使っている辺の集合とする．任意の列  $j$  について， $R_j$  中の任意の頂点と  $R_{j+1}$  中の任意の頂点との間の  $H$  における最短パスは，辺  $(v_{1,j}, v_{1,j+1})$  が  $Z$  に含まれていないときは少なくとも  $K$  以上になる． $|Z| \leq \frac{D-K}{2}$  であることより，このとき  $H$  における頂点  $v_{z,0}$  と  $v_{z,(D-K)K}$  の間の最短パスの長さは少なくとも  $\frac{(D-K)K}{4}$  以上となる． $D > 2K$  より  $D - K > D/2$  なので，このとき上述のパスの長さは  $\Omega(kD)$  となり背理法の仮定に矛盾する．よって補題が証明された．□

定理 3.2 は補題 3.6 および補題 3.5 より直ちに得られる．

#### 4. まとめと今後の課題

本稿では，コードグラフの一般化グラフである  $k$ -コードグラフに対して，クオリティが  $O(kD)$  となる低競合ショートカットの存在性とそれを 1 ラウンドで構成する分散アルゴリズムを示すとともに，任意のショートカットのクオリティが  $\Omega(kD)$  となるような  $k$ -コードグラフのインスタンスが存在することにより，提案手法の最適性を示した．今後の課題として，提案したショートカット構成手法の他のグラフクラスへの適用とその解析を挙げるができる．

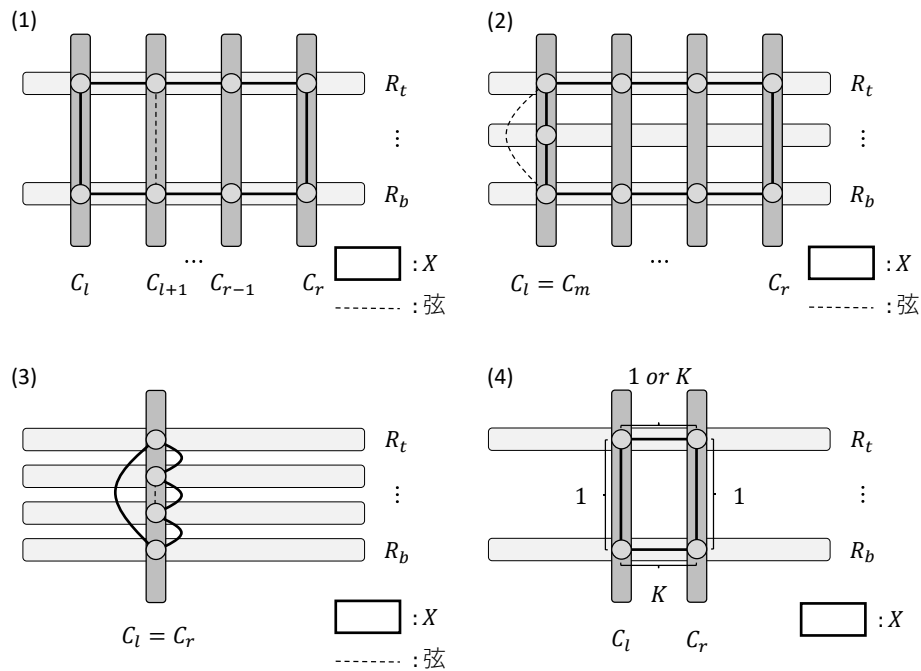


図 5 補題 3.5 の証明

### 謝辞

本研究は JST 戦略的国際共同研究プログラム (SICORP) ならびに JSPS 科研費 16H02878 の支援を受けました。

### 参考文献

[1] Ghaffari, M. and Haeupler, B.: Distributed algorithms for planar networks ii: Low-congestion shortcuts, mst, and min-cut, *Proceedings of the twenty-seventh annual ACM-SIAM symposium on Discrete algorithms*, SIAM, pp. 202–219 (2016).

[2] Gallager, R. G., Humblet, P. A. and Spira, P. M.: A distributed algorithm for minimum-weight spanning trees, *ACM Transactions on Programming Languages and systems (TOPLAS)*, Vol. 5, No. 1, pp. 66–77 (1983).

[3] Peleg, D. and Rubinfeld, V.: A near-tight lower bound on the time complexity of distributed minimum-weight spanning tree construction, *SIAM Journal on Computing*, Vol. 30, No. 5, pp. 1427–1442 (2000).

[4] Elkin, M.: Distributed approximation: a survey, *ACM SIGACT News*, pp. 40–57 (2004).

[5] Sarma, A. D., Holzer, S., Kor, L., Korman, A., Nanongkai, D., Pandurangan, G., Peleg, D. and Wattenhofer, R.: Distributed verification and hardness of distributed approximation, *SIAM Journal on Computing*, Vol. 41, No. 5, pp. 1235–1265 (2012).

[6] Kutten, S. and Peleg, D.: Fast distributed construction of small-k-dominating sets and applications, *Journal of Algorithms*, Vol. 28, No. 1, pp. 40–66 (1998).

[7] Garay, J. A., Kutten, S. and Peleg, D.: A sublinear time distributed algorithm for minimum-weight spanning trees, *SIAM Journal on Computing*, Vol. 27, No. 1, pp. 302–316 (1998).

[8] Haeupler, B., Izumi, T. and Zuzic, G.: Near-optimal low-congestion shortcuts on bounded parameter graphs, *International Symposium on Distributed Computing*, Springer, pp. 158–172 (2016).

[9] Haeupler, B., Li, J. and Zuzic, G.: Minor excluded net-

work families admit fast distributed algorithms, *arXiv preprint arXiv:1801.06237* (2018).

[10] Haeupler, B. and Li, J.: Faster Distributed Shortest Path Approximations via Shortcuts, *32nd International Symposium on Distributed Computing (DISC 2018)*, Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik (2018).

[11] Ghaffari, M. and Parter, M.: Near-Optimal Distributed DFS in Planar Graphs, *Proc. of 31st International Symposium on Distributed Computing (DISC)*, Vol. 91, pp. 21:1–21:16 (2017).

[12] Ghaffari, M.: Near-optimal scheduling of distributed algorithms, *Proceedings of the 2015 ACM Symposium on Principles of Distributed Computing*, ACM, pp. 3–12 (2015).