

Title	バンド幅問題の効率のよいアルゴリズムの開発に関する研究
Author(s)	中西, 朗裕
Citation	
Issue Date	2010-03
Type	Thesis or Dissertation
Text version	author
URL	http://hdl.handle.net/10119/8965
Rights	
Description	Supervisor:上原隆平, 情報科学研究科, 修士

修士論文

バンド幅問題の
効率のよいアルゴリズムの開発に関する研究

北陸先端科学技術大学院大学
情報科学研究科情報科学専攻

中西 朗裕

2010年3月

修士論文

バンド幅問題の
効率のよいアルゴリズムの開発に関する研究

指導教官 上原隆平 准教授

審査委員主査 上原隆平 准教授
審査委員 浅野哲夫 教授
審査委員 宮地充子 教授

北陸先端科学技術大学院大学
情報科学研究科情報科学専攻

0810044 中西 朗裕

提出年月: 2010年2月

概要

バンド幅問題とは、与えられたグラフの頂点を一列に並べたときの隣接している頂点間の距離の最大値を最小化する問題である。この問題は、疎行列の計算や、分子生物学に応用を持つことが知られている。

バンド幅問題は、一般のグラフに対して NP 完全であるだけでなく、グラフを木に限定しても NP 完全である。しかし、いくつかのグラフクラスでバンド幅問題を解く多項式時間アルゴリズムが提案されている。

本論文では、チェーングラフ上のバンド幅問題を解くアルゴリズムを精細に調べ、その性質を明らかにする。この結果を利用し、2部パーミュテーショングラフ上のバンド幅問題を解く既存の $O(n^2)$ 時間アルゴリズムを改善し、線形時間アルゴリズムを提案する。

目次

第1章	はじめに	1
1.1	研究の背景	1
1.2	研究の目的	1
1.3	本論文の流れ	2
第2章	準備	3
2.1	グラフの基本用語	3
2.1.1	グラフとは	3
2.1.2	次数	3
2.1.3	隣接点集合	4
2.1.4	twin	4
2.1.5	完全グラフ	4
2.1.6	誘導部分グラフ	4
2.1.7	クリーク	5
2.1.8	独立点集合	5
2.2	グラフクラス	5
2.2.1	2部グラフ	5
2.2.2	パーミュテーショングラフ	5
2.2.3	2部パーミュテーショングラフ	6
2.2.4	チェーングラフ	7
2.2.5	区間グラフ	9
2.2.6	真区間グラフ	10
2.2.7	閾値グラフ	11
第3章	バンド幅	12
3.1	バンド幅	12
3.2	最適なレイアウト	13
3.3	バンド幅問題	14
3.4	proper interval completion	14
3.4.1	minimum proper interval completion	14
3.4.2	minimum proper interval completion とバンド幅の関係	14

第4章	アルゴリズム	16
4.1	閾値グラフ上のバンド幅問題を解くアルゴリズム	16
4.2	チェーングラフ上のバンド幅問題	18
4.2.1	wiper	18
4.2.2	チェーングラフの wiper による区間表現	18
4.2.3	チェーングラフ上のバンド幅問題を解くアルゴリズム	19
4.2.4	チェーングラフの最適なレイアウトの性質	21
4.3	2部パーミュテーショングラフ上の k バンド幅問題	22
4.3.1	2部パーミュテーショングラフの最適なレイアウトの持つ性質	22
4.3.2	アルゴリズム	23
4.3.3	アルゴリズムの説明	24
4.3.4	データ構造	24
4.3.5	計算時間	25
第5章	おわりに	27

第1章 はじめに

1.1 研究の背景

グラフ $G=(V, E)$ のレイアウト σ とは V から集合 $\{1, 2, \dots, |V|\}$ への全単射である。グラフ G のレイアウト σ のバンド幅 $bw(\sigma)$ とは, $\max\{|\sigma(u) - \sigma(v)| \mid \{u, v\} \in E\}$ である。グラフ G のバンド幅 $bw(G)$ とは, $\min_{\sigma}\{bw(\sigma)\}$ である。直感的には, グラフ G のバンド幅 $bw(G)$ を求めることは, 与えられたグラフの頂点を一列に並べたとき, 隣接している2頂点の中で最も離れたものの距離が最小である頂点の配置を求めることである。

バンド幅問題とは, 入力としてグラフ G が与えられたとき, $bw(G)$ を出力する問題である。 k バンド幅問題とは, 入力としてグラフ G と定数 k が与えられたときに, バンド幅 k 以下のレイアウトが存在するかを問う問題である。

バンド幅問題は疎行列の計算や, 分子生物学に応用を持つことが知られている [1] [2]。

バンド幅問題は, 一般のグラフに対して NP 完全であり, グラフを木に限定しても NP 完全である [3] [4]。一般のグラフに対してバンド幅問題を解く厳密なアルゴリズムは, 2008年に Cygan と Pilipczuk により $O(5^n)$ 時間のアルゴリズムが提案された [5]。しかしグラフクラスを限定すると, いくつかのグラフクラスに対してバンド幅問題または k バンド幅問題を解く多項式時間アルゴリズムが提案されている。閾値グラフとチェーングラフではバンド幅問題を解く線形時間のアルゴリズムが提案されている [6]。また, 区間グラフは $O(n \log n)$ 時間, 2部パーミュテーショングラフは $O(n^2)$ 時間の k バンド幅問題を解くアルゴリズムが提案されている [6] [7]。

1.2 研究の目的

本研究の目的は, 2部パーミュテーショングラフの k バンド幅問題を解く $O(n^2)$ 時間のアルゴリズムの計算時間の改善である。

2部パーミュテーショングラフはチェーングラフの列で表現することができる。2部パーミュテーショングラフの k バンド幅問題を解くアルゴリズムは, この性質を利用している。このため, チェーングラフ上のバンド幅問題を解くアルゴリズムを精細に調べ, その性質を明らかにする。これを利用して, 2部パーミュテーショングラフの k バンド幅問題を解く既存のアルゴリズムの線形時間への改善を行う。

1.3 本論文の流れ

本論文では，第 2 章において，本論文で扱うグラフ理論の基本的な用語の定義と，グラフクラスの定義と性質について述べる．第 3 章において，バンド幅問題の定義と，proper interval completion について述べる．第 4 章において，閾値グラフとチェーングラフのアルゴリズムの概要について述べ，2 部パーミュテーショングラフの k バンド幅問題を解く既存のアルゴリズムの線形時間への改善を示す．第 5 章をまとめとする．

第2章 準備

本章では、グラフに関する基本的な用語と、いくつかのグラフクラスについて説明する。

2.1 グラフの基本用語

2.1.1 グラフとは

グラフ $G=(V, E)$ は、頂点集合 V と辺集合 E からなる。辺集合 E は頂点集合 V の2元部分集合である。頂点 $v_i, v_j \in V$ に対して辺 $e \in E$ が存在して、 $e = \{v_i, v_j\}$ であるとき、 v_i と v_j は隣接しているという。図 2.1 に例を示す。この例では頂点集合 V は $\{a, b, c, d, e, f\}$ 、辺集合 E は $\{\{a, b\}, \{a, c\}, \{b, c\}, \{b, d\}, \{b, e\}, \{c, d\}, \{c, e\}, \{d, e\}, \{d, f\}\}$ である。

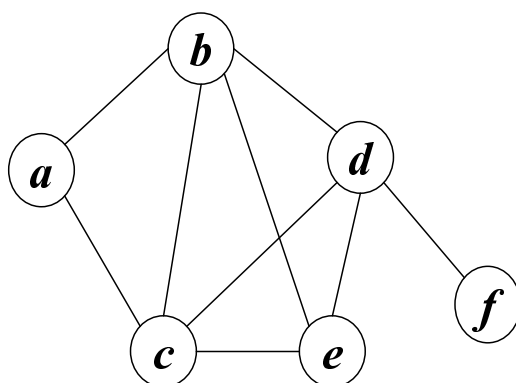


図 2.1: グラフの例

2.1.2 次数

グラフ $G = (V, E)$ において、頂点 $v \in V$ の次数とは v に隣接する頂点の数である。 v の次数を $d(v)$ と表す。図 2.1 を使って例を示す。 $d(b) = 4$ 、 $d(f) = 1$ である。

2.1.3 隣接点集合

グラフ $G = (V, E)$ のある頂点 $v \in V$ の隣接点集合とは, $N(v) = \{u \in V | \{u, v\} \in E\}$ を満たす頂点集合である. 図 2.1 を使って例を示す. $N(b) = \{a, c, d, e\}$ である.

2.1.4 twin

グラフ $G = (V, E)$ の頂点 $u, v \in V$ が twin であるとは, $N(u) = N(v)$ を満たすことである.

2.1.5 完全グラフ

グラフ $G = (V, E)$ の任意の 2 頂点が隣接しているとき, G を完全グラフという. 図 2.2 は $|V| = 5$ の完全グラフの例である.

2.1.6 誘導部分グラフ

グラフ $G = (V, E)$ が与えられたとき, $V' \subseteq V$ に対して $E' = \{\{v_i, v_j\} \in E | v_i, v_j \in V'\}$ とする. このとき, $G' = (V', E')$ を G の V' による誘導部分グラフと呼び, $G'[V']$ と表す. 与えられたグラフ $G = (V, E)$ を図 2.1 とする. 誘導部分グラフの例を図 2.3 に示す. これは, $V' = \{a, b, d, e\}$ としたときの, V' による誘導部分グラフ $G' = [V']$ である.

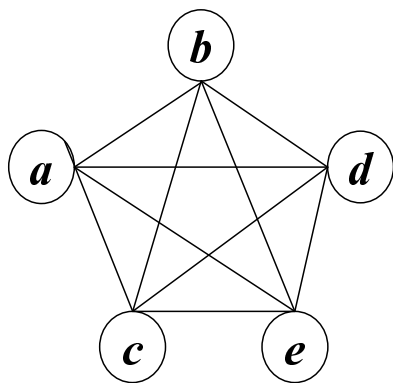


図 2.2: 完全グラフの例

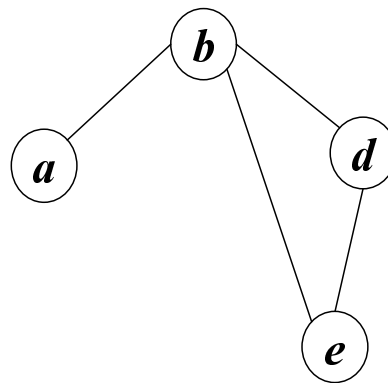


図 2.3: 図 2.1 の誘導部分グラフ G' の例

2.1.7 クリーク

グラフ $G = (V, E)$ の誘導部分グラフ $G' = (V', E')$ が完全グラフであるとき, V' をクリークという. G の中で頂点数最大のクリークを最大クリークという. 与えられたグラフ $G = (V, E)$ を図 2.1 として, 例を示す. $V' = \{b, c, d, e\}$ とすると, V' による誘導部分グラフ $G' = (V', E')$ は完全グラフである. よって, $V' = \{b, c, d, e\}$ はクリークである.

2.1.8 独立点集合

グラフ $G = (V, E)$ が与えられたとき, $V' \subseteq V$ に属する任意の 2 頂点が隣接しないとき, V' を G の独立点集合という.

2.2 グラフクラス

2.2.1 2部グラフ

2部グラフとは, グラフ $G = (V, E)$ において V を 2 つの頂点集合 X, Y に分割することができ, すべての辺が X, Y の間を結ぶものに限られるグラフである. 2部グラフは $G = (X, Y, E)$ とも表される. 図 2.4 に例を示す. この例では $X = \{x_1, x_2, x_3, x_4, x_5\}$, $Y = \{y_1, y_2, y_3, y_4\}$ である.

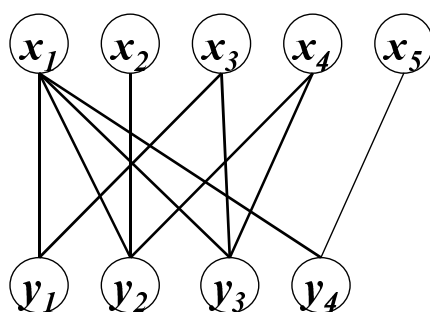


図 2.4: 2部グラフの例

2.2.2 パーミュテーショングラフ

グラフ $G = (V, E)$, ただし $V = \{v_1, v_2, \dots, v_n\}$ がパーミュテーショングラフであるとは, $\{v_i, v_j\} \in E$ であることの必要十分条件が $(i - j)(\sigma(v_i) - \sigma(v_j)) \leq 0$ であるような V 上の順列 σ が存在することである.

直観的に説明すると, パーミュテーショングラフとはライン表現を持つグラフである. ライン表現とは, 2本の平行線 L_1, L_2 に端点を持つ線分の集合である. 各線分はグラフの

各頂点に対応し，グラフが辺を持つ必要十分条件は対応する線分が交差することである．パーミュテーショングラフの例を図 2.5 に示す．図 2.5 のライン表現を図 2.6 に示す．図 2.5 の頂点 1 は図 2.6 の両端に 1 を持つ線分に対応している．たとえば，図 2.5 の頂点 1 と 6 は隣接していて，図 2.6 で端点に 1 を持つ線分と 6 を持つ線分は交差している．一方，図 2.5 の頂点 2 と 5 は隣接しておらず，図 2.6 で端点に 2 を持つ線分と 5 を持つ線分は交差していない．

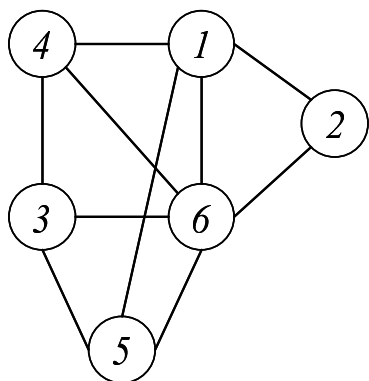


図 2.5: パーミュテーショングラフの例

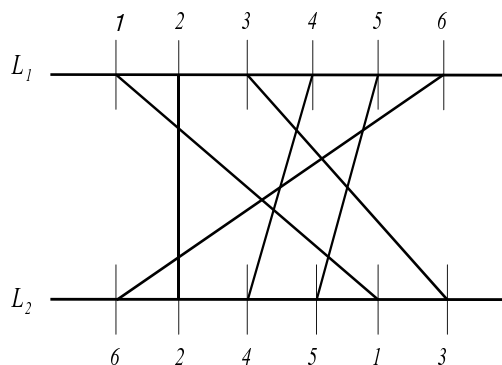


図 2.6: 図 2.5 のライン表現

2.2.3 2部パーミュテーショングラフ

グラフ G が 2 部グラフかつパーミュテーショングラフであるとき， G は 2 部パーミュテーショングラフであるという．2 部パーミュテーショングラフの例を図 2.7 に示す．図 2.8 は図 2.7 のライン表現である．

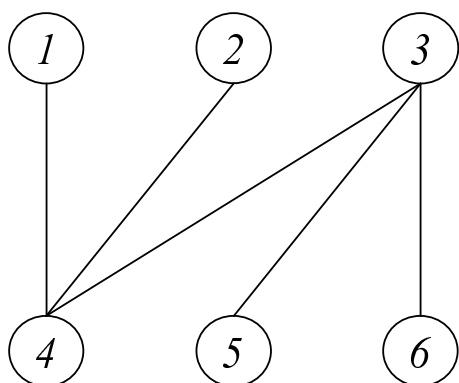


図 2.7: 2部パーミュテーショングラフの例

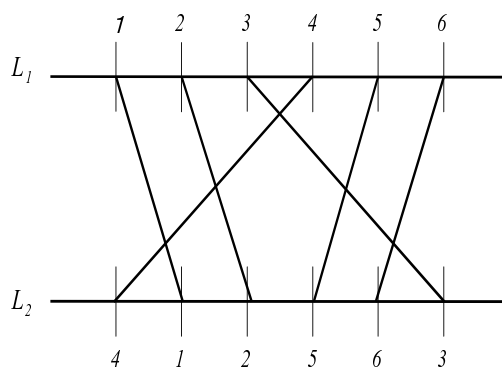


図 2.8: 図 2.7 のライン表現

2.2.4 チェーングラフ

$G = (X, Y, E)$ を 2 部グラフとする . ただし $X = \{x_1, x_2, \dots, x_n\}$, $Y = \{y_1, y_2, \dots, y_{n'}\}$ とする . σ を X の順序とする . 任意の $y \in Y$ について $N(y)$ が σ 上の連結する X の要素からなるとき , σ は adjacency property を持つという . 2 部グラフ $G = (X, Y, E)$ がチェーングラフであるとは , adjacency property を満たす X と Y の順序が存在し , かつ X の順序が $N(x_n) \subseteq N(x_{n-1}) \subseteq \dots \subseteq N(x_2) \subseteq N(x_1)$ を満たすことである . (このとき , $N(y_m) \subseteq N(y_{m-1}) \subseteq \dots \subseteq N(y_2) \subseteq N(y_1)$ も成立する .)

図 2.9 に例を示す . この例では , $N(x_5) \subseteq N(x_4) \subseteq N(x_3) \subseteq N(x_2) \subseteq N(x_1)$ である .

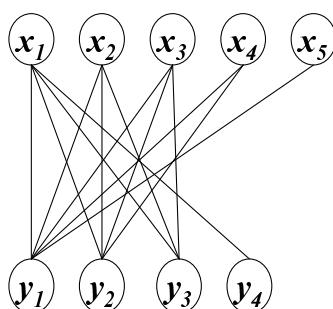


図 2.9: チェーングラフの例

チェーングラフは二次元平面上のある交差モデルを持つ . チェーングラフの交差モデルは , x 軸に平行な線分の集合 $\{H_1, H_2, \dots, H_n\}$, y 軸に平行な線分の集合 $\{V_1, V_2, \dots, V_{n'}\}$ で表現される . 線分 $H_i (1 \leq i \leq n)$ は頂点 $x_i (1 \leq i \leq n)$ と対応する . 線分 $V_i (1 \leq i \leq n')$ は頂点 $y_i (1 \leq i \leq n')$ と対応する . すべての x 軸に平行な線分の右側の端点は , y 軸に平行なある直線上にある . すべての y 軸に平行な線分の下側の端点は , x 軸に平行なある直線上にある . x 軸に平行な線分 $\{H_1, H_2, \dots, H_n\}$ の左端の x 座標は単調増加する . y 軸に平行な線分 $\{V_1, V_2, \dots, V_{n'}\}$ の上端の y 座標は単調増加する . チェーングラフ上の 2 頂点間に辺が存在する必要十分条件は各頂点に対応する 2 つの線分が交差することである . 図 2.9 の交差モデルの例を図 2.10 に示す . たとえば , 頂点 x_1 と y_3 は隣接していて , 線分 H_1 と V_3 は交差を持つ . 一方 , 頂点 x_3 と y_4 は隣接しておらず , 線分 H_3 と V_4 は交差を持たない .

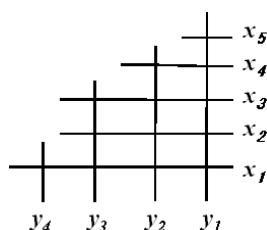


図 2.10: 図 2.9 の交差モデル

補題 1. [8] チェーングラフは 2 部パーミュテーショングラフである .

補題 1 を証明する . チェーングラフの交差モデルから , 2 部パーミュテーショングラフのライン表現を作ることができる . X と Y の隣接関係を壊さずに図 2.11 の L_1, L_2 を直線状に引き延ばすと , 図 2.12 のライン表現が得られる . なお , このとき L_1 上の X と Y が分離されていることに注意する .

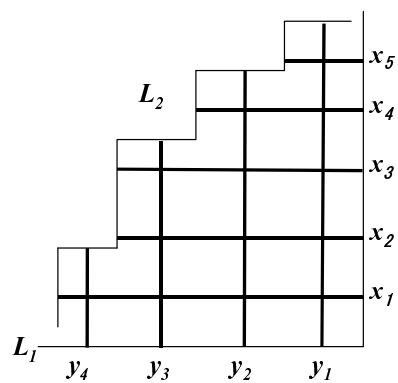


図 2.11: 交差モデル

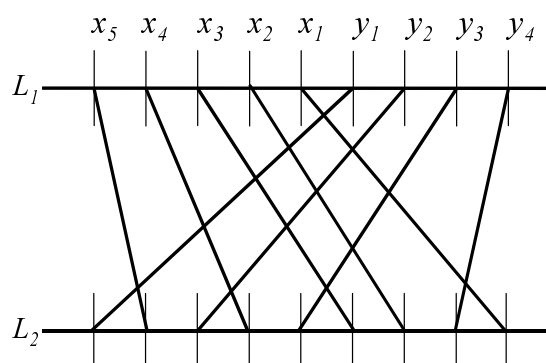


図 2.12: 図 2.11 のライン表現

補題 2. [8] G を 2 部パーミュテーショングラフとする . L_1, L_2 を G のライン表現の 2 本の平行線とする . $V'_0 = \{x_0 \mid x_0 \text{ は } L_1 \text{ 上の最も左の端点に対応する頂点}\}$ とする . $V'_i = \{v \mid \text{dist}(x_0, v) = i\}$ とする . $V_i = V'_i (i = 1, 3, 4, \dots, m)$, $V_2 = V'_0 \cup V'_2$ とする . このとき , G_i を $G_i = G[V_i \cup V_{i+1}] = G(V_i \cup V_{i+1}, E_i) (i = 1, 2, \dots, m)$ と定義する . ただしここで E_i は G_i の辺集合である . グラフは G_1 から始める . このとき , G_1, G_2, \dots, G_m はチェーングラフである . しかも , 任意の *twin* $\{u, w\}$ に対して , $u \in V_i$ ならば $w \in V_i$ である .

例を図 2.13 , 2.14 , 2.15 に示す . 与えられた 2 部パーミュテーショングラフ G を図 2.13 とする . $V_1 = \{1, 3, 7\}$, $V_2 = \{2, 4, 5, 6\}$, $V_3 = \{8, 9, 10\}$ である . 図 2.14 は $G_1 = (V_1 \cup V_2, E_1)$ である . 図 2.15 は $G_2 = (V_2 \cup V_3, E_2)$ である .

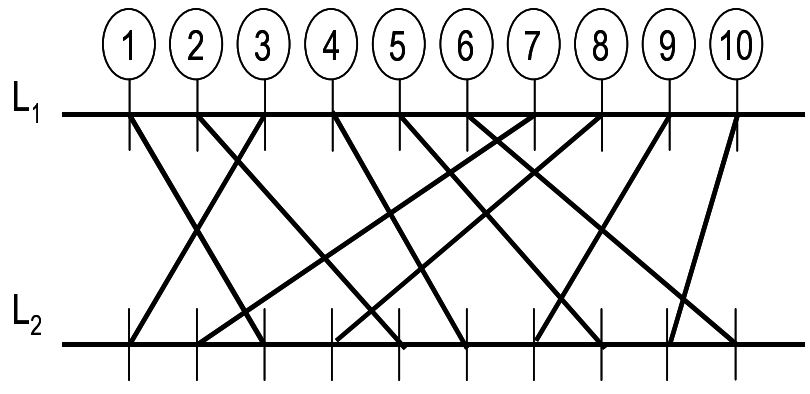


図 2.13: 2 部パーミュテーショングラフのライン表現の例

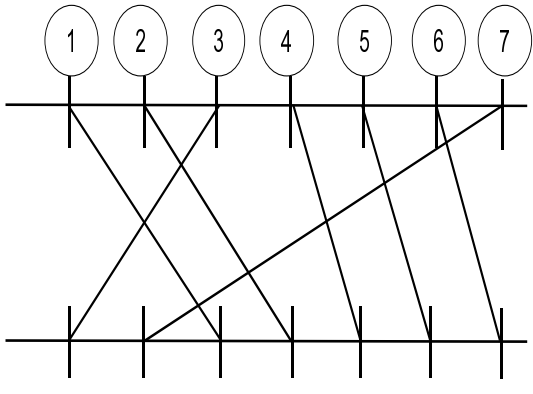


図 2.14: チェーングラフ G_1

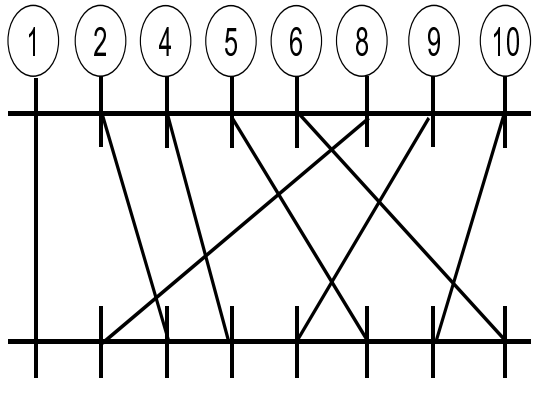


図 2.15: チェーングラフ G_2

2.2.5 区間グラフ

区間表現を持つグラフを区間グラフという . グラフ $G = (V, E)$ の区間表現とは , 数直線上の区間の集合 \mathcal{I} である . V の各頂点は \mathcal{I} の各区間と 1 対 1 対応する . G において各

頂点間に辺が存在することの必要十分条件は、対応する区間同士が重なりを持つことである。 $v_i \in V$ に対応する区間 I_{v_i} の左端点を $L(I_{v_i})$ 、右端点を $R(I_{v_i})$ とする。図 2.16 を与えられたグラフ G としたとき、図 2.17 は対応する区間表現の例である。図 2.16 で頂点 a と b は隣接し、図 2.17 の区間 I_a と I_b は重なりを持つ。一方、図 2.16 で頂点 b と d は隣接しておらず、図 2.17 の区間 I_b と I_d は重なりを持たない。

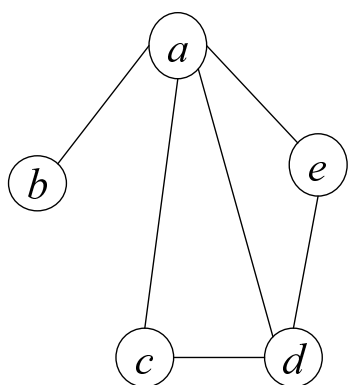


図 2.16: 区間グラフの例

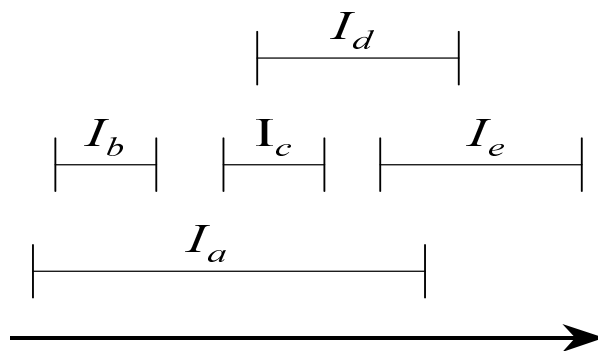


図 2.17: 図 2.16 の区間表現

2.2.6 真区間グラフ

真区間表現を持つグラフ G を真区間グラフという。真区間表現とは、区間表現で、どの区間も別の区間に真に含まれることはないものである。

図 2.18 は真区間グラフ G の例であり、図 2.19 が対応する区間表現である。

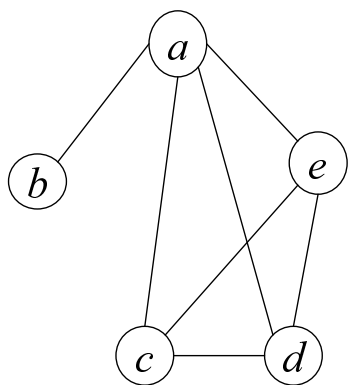


図 2.18: 真区間グラフの例

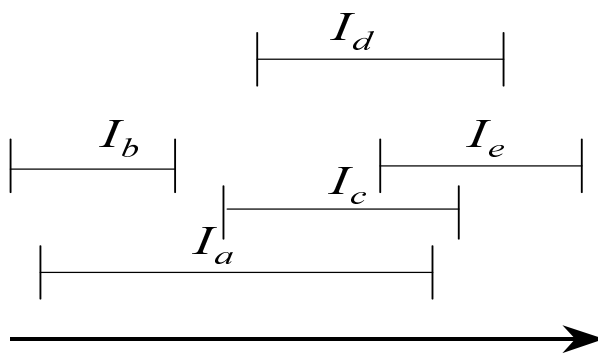


図 2.19: 図 2.18 の真区間表現

2.2.7 閾値グラフ

グラフ $G = (V, E)$ が以下の条件を満たすとき, G を閾値グラフという. すべての頂点 $v \in V$ に対する非負の重み $w(v)$ と, ある閾値 t が存在し, G において頂点間 $u, v \in V$ に辺が存在することの必要十分条件は, $w(u) + w(v) \geq t$ を満たすことである.

図 2.20 に閾値グラフ $G = (V, E)$ の例を示す. $V = \{v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8\}$ である. 頂点の重みは, $w(v_1) = 1, w(v_2) = 2, w(v_3) = 3, w(v_4) = 3, w(v_5) = 4, w(v_6) = 4, w(v_7) = 4, w(v_8) = 5$ である. 閾値 $t = 6$ である. $w(v_1) + w(v_8) \geq 6$ であるので, v_1 と v_8 は隣接している. 一方, $w(v_1) + w(v_4) < 6$ でないので, v_1 と v_4 は隣接していない.

閾値グラフは区間表現を持つ. したがって, 閾値グラフは区間グラフである. 以下ではこれを説明する.

$V = \{v_1, v_2, \dots, v_n\}$ とする. ここで一般性を失うことなく $w(v_1) \leq w(v_2) \leq \dots \leq w(v_n)$ と仮定することができる. このとき, $w(v_{i-1}) + w(v_i) < t$ かつ $w(v_i) + w(v_{i+1}) \geq t$ を満たす v_i について以下が成立する. $V' = \{v_1, v_2, \dots, v_i\}$ は独立点集合であり, $V'' = \{v_{i+1}, v_{i+2}, \dots, v_n\}$ はクリークである.

V' の頂点 $\{v_1, v_2, \dots, v_i\}$ に対応する区間を $[1, 1], [2, 2], \dots, [i, i]$ とする. $v_j \in V'' (j = i+1, \dots, n)$ に対応する区間を $[m_j, i]$ とする. ここで m_j は G において最 v_j の隣接する頂点の中で最小の添字を持つ頂点の添字である. このとき, こうして得られた区間の集合は, 明らかに G の区間表現になっている.

図 2.20 を閾値グラフ G の例とし, 図 2.21 に区間表現の例を示す. $V' = \{v_1, v_2, v_3, v_4\}$ はどの 2 頂点間にも辺がない. $V'' = \{v_5, v_6, v_7, v_8\}$ はクリークである.

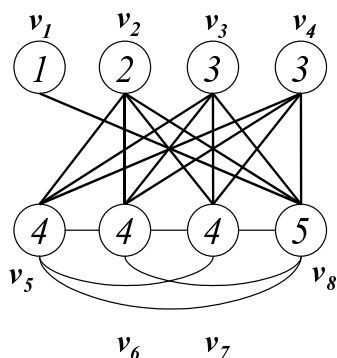


図 2.20: 閾値グラフの例

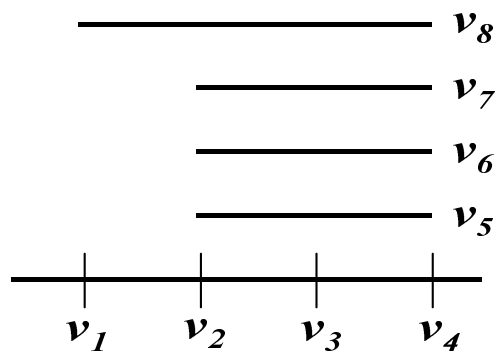


図 2.21: 図 2.20 の区間表現

第3章 バンド幅

本章では、バンド幅の説明をした後、バンド幅問題の定義を述べる。さらに、バンド幅と proper interval completion の関係について述べる。

3.1 バンド幅

本節では、グラフ $G = (V, E)$ のレイアウトと、そのレイアウトにおけるバンド幅について述べる。

グラフ $G = (V, E)$ のレイアウト σ とは V から集合 $\{1, 2, \dots, |V|\}$ への全単射である。すると σ は V の頂点集合の順序付けとみなすことができる。グラフ G のレイアウト σ のバンド幅 $bw(\sigma)$ とは、 $\max\{|\sigma(u) - \sigma(v)| \mid \{u, v\} \in E\}$ である。

図 3.1 にグラフ $G = (V, E)$ の例を示す。

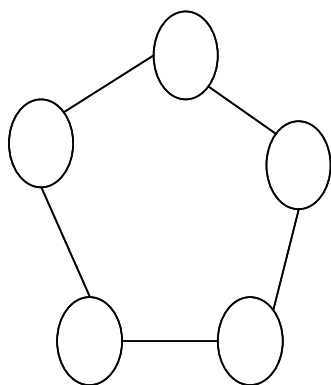


図 3.1: グラフ G

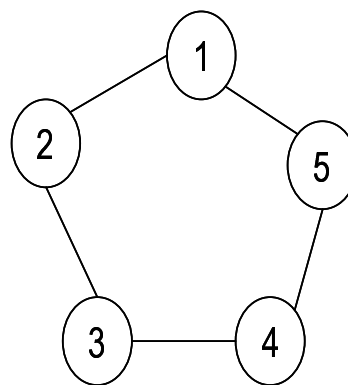


図 3.2: レイアウト σ

この G に対する、レイアウト σ の例を図 3.2 に示す。

グラフ G の頂点集合 V を図 3.2 の σ の順序に基づいて一列に並べたものが、図 3.3 である。このとき、隣接している 2 頂点で最も離れているものは、1 と 5 である。よって、このレイアウト σ では、 $bw(\sigma) = 4$ である。

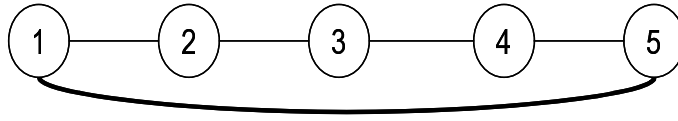


図 3.3: 図 3.2 の順序に基づいて頂点を一列に並べたグラフ

3.2 最適なレイアウト

グラフ G のバンド幅とは, $bw(G) = \min_{\sigma} \{bw(\sigma)\}$ である. バンド幅が $bw(G)$ であるレイアウトを G の最適なレイアウトという. 図 3.1 における最適なレイアウトを図 3.4 に示す.

図 3.5 はグラフ G の頂点集合 V を図 3.4 のレイアウトの順序通りに並べたものである. このとき, 隣接している 2 頂点で最も離れているものは, 1 と 3, 2 と 4, 3 と 5 である. このレイアウトでは, バンド幅は 2 である. また, $bw(\sigma) = 1$ を実現できるレイアウト σ は存在しないので, グラフ G のバンド幅 $bw(G) = 2$ である.

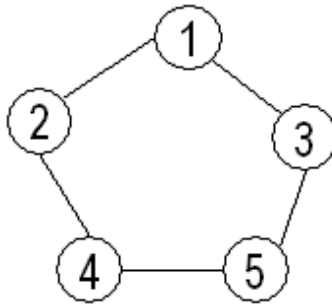


図 3.4: 最小のバンド幅を持つレイアウト

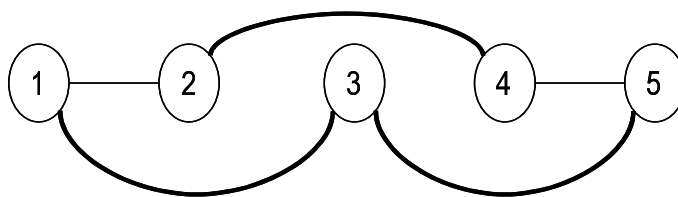


図 3.5: 図 3.4 に示した順序に基づいて頂点を一列に並べたグラフ

3.3 バンド幅問題

バンド幅問題とは，入力としてグラフ $G = (V, E)$ が与えられたときに， $bw(G)$ を出力する最適化問題である．

k バンド幅問題とは，入力としてグラフ G と定数 k が与えられたときに，バンド幅 k 以下のレイアウトが存在するかを問う判定問題である．

3.4 proper interval completion

与えられたグラフ $G = (V, E)$ について， G の proper interval completion とは，真区間グラフ $G' = (V, E')$ で $E \subseteq E'$ を満たすものである．

例を示す．図 2.16 を与えられたグラフ $G = (V, E)$ とする．このグラフ G は真区間表現を持たない． $E' = E \cup \{c, e\}$ とし，グラフ $G' = (V, E')$ とすると，図 2.18 の真区間グラフ G' が得られる．これは， G の proper interval completion である．

3.4.1 minimum proper interval completion

グラフ G の proper interval completion のうち，最大クリークの大きさが最小のものを G の minimum proper interval completion という．

3.4.2 minimum proper interval completion とバンド幅の関係

補題 3. [9] G を連結で *twin* のない真区間グラフとし， \mathcal{I} を G の真区間表現とする．すると， G の任意の真区間表現 \mathcal{I}' について， $L_{(I_{v_1})} < L_{(I_{v_2})} < \cdots < L_{(I_{v_n})} \rightarrow L_{(I'_{v_1})} < L_{(I'_{v_2})} < \cdots < L_{(I'_{v_n})}$ または， $L_{(I_{v_n})} < L_{(I_{v_{n-1}})} < \cdots < L_{(I_{v_1})}$ が成り立つ．

補題 4. [10] $G' = (V, E')$ を与えられたグラフ $G = (V, E)$ の *minimum proper interval completion* とする． $\mathcal{I}(G') = (I'_{v_1}, I'_{v_2}, \dots, I'_{v_n})$ を， G' の真区間表現とする．そのとき，順序 v_1, v_2, \dots, v_n は G の最適なレイアウトを与える．また， G' の最大クリークを C とすると， $bw(G) = |C| - 1$ である．

与えられたグラフ $G = (V, E)$, $V = \{a, b, c, d, e, f, g\}$ とする. G の minimum proper interval completion を $G' = (V, E')$ とする. G' の真区間表現を図 3.6 とする. このとき, 補題 3, 4 より, G の最適なレイアウトは a, b, c, d, e, f, g であり, $bw(G) = 2$ である.

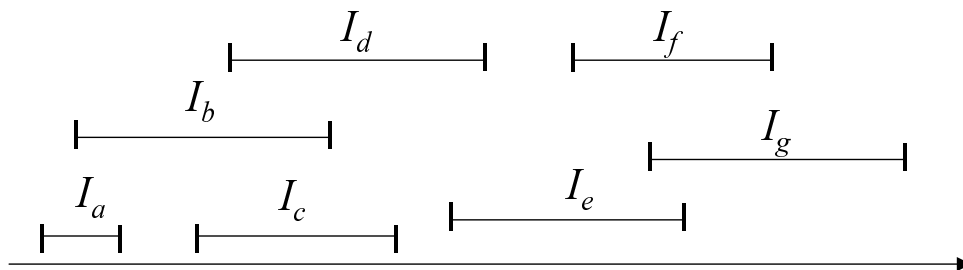


図 3.6: チェーングラフの区間表現の例

第4章 アルゴリズム

本章では、2部パーミュテーショングラフ上の k バンド幅問題を解く線形時間アルゴリズムを提案する。そのために、まず、閾値グラフ上のバンド幅問題を解くアルゴリズムと、チェーングラフ上のバンド幅問題を解くアルゴリズムの概要を説明する。そして、チェーングラフの最適なレイアウトの性質と、2部パーミュテーショングラフの最適なレイアウトの持つ性質について述べる。

4.1 閾値グラフ上のバンド幅問題を解くアルゴリズム

定理 1. [6] $G = (V, E)$ を連結な閾値グラフとする。 $|V| = n$ とする。 G の区間表現 $\mathcal{I}(G)$ が与えられたとする。このとき、 $bw(G)$ は $O(n)$ 時間で計算できる。

ここでは、閾値グラフ上のバンド幅問題を解くアルゴリズムの概要について述べる。

与えられた閾値グラフを $G = (V, E)$ とする。 G の頂点集合 V は独立点集合である V' とクリークである V'' の2つに分割される。 $V' = \{v_1, v_2, \dots, v_{l-1}, v_l\}$, $V'' = \{v_{l+1}, v_{l+2}, \dots, v_{n-1}, v_n\}$ とする。 G の区間表現を $\mathcal{I}(G)$ とする。 G の minimum proper interval completion G' を求めることにより、最適なレイアウトが得られる。ここでは、区間表現 $\mathcal{I}(G)$ 上での minimum proper interval completion を考える。まず、前処理として、すべて頂点の次数を計算し、頂点をインデックスに持つ配列に格納する。これは $\mathcal{I}(G)$ のすべての区間上の両端点を順に調べることにより得られる。したがって、この配列は $O(n)$ 時間で計算できる。次に、 $\{v_1, v_2, \dots, v_{l-1}\}$ に対応するすべての区間 $\{I_{v_1}, I_{v_2}, \dots, I_{v_{l-1}}\}$ の $R(I_{v_i})$ ($1 \leq i \leq l-1$) を l にする。このとき、 $[l, l]$ でサイズ $|V|$ のクリークができる。いま、minimum proper interval completion を求めたい。そこで、 I_{v_i} ($1 \leq i \leq l-1$) について、 $I_{v_1}, I_{v_2}, \dots, I_{v_{l-1}}$ の順に $L(I_{v_i})$ を 1 にする。この $L(I_{v_i})$ を 1 にする操作をフリップという。フリップを続けることで、最大クリークが最小の区間表現を得ることができる。最大クリークの候補は、 $(1)\{v_1, v_2, \dots, v_m\} \cup \{v_{l+1}, \dots, v_n\}$, $(2)d(v_m)$, $(3)\{v_{m+1}\} \cup \{v_{l+1}, \dots, v_n\}$ ($m(1 \leq i \leq l)$) の内のどれかである。この m は、フリップを続けることで見つけることができる。フリップを続けることで得られた区間表現から真区間表現 $\mathcal{I}'(G)$ を得ることができる。補題 4 より、最適なレイアウトは、真区間表現 $\mathcal{I}'(G)$ から得ることができる。 $\mathcal{I}(G)$ の例を図 4.1 に、 $\mathcal{I}'(G)$ の例を図 4.2 に示す。

計算時間を考える。(1)は v_1 をフリップしたときの計算のときに配列を参照する。これは定数時間でできる。また、1回のフリップについて、クリークのサイズが1増えるだけ

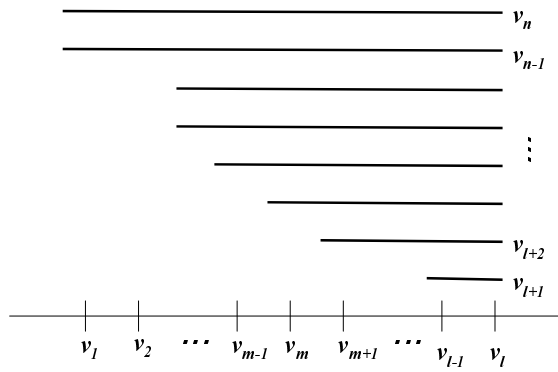


図 4.1: $\mathcal{I}(G)$ の例

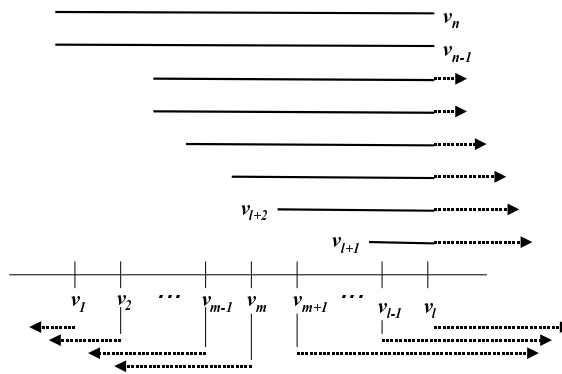


図 4.2: $\mathcal{I}'(G)$ の例

なので、クリークのサイズは定数時間で計算できる。(2) は配列を参照するだけなので定数時間である。(3) は1回のフリップについて、サイズが1減るだけなので定数時間で計算できる。すべての区間について高々定数時間しか計算を必要としない。よって、真区間表現 $I'(G)$ を得るための計算時間は $O(n)$ 時間である。前処理の $O(n)$ 時間と合わせて、全体の計算時間は $O(n)$ 時間である。

4.2 チェーングラフ上のバンド幅問題

2部パーミュテーショングラフ上の k バンド幅問題を解くアルゴリズムは、チェーングラフの最適なレイアウトの性質を利用する。そのために、本節ではチェーングラフ上のバンド幅問題を解くアルゴリズムの概要について述べる。そして、チェーングラフの最適なレイアウトの性質について述べる。

定理 2. [6] $G = (V, E)$ をチェーングラフとする。 $|V| = n$ とする。このとき、 $bw(G)$ は $O(n)$ 時間で計算できる。

4.2.1 wiper

チェーングラフ上のバンド幅問題を解くアルゴリズムの説明のために $wiper(x_i)$ について述べる。

$G = (X, Y, E)$ をチェーングラフとする。 $X = \{x_1, x_2, \dots, x_n\}, Y = \{y_1, y_2, \dots, y_{n'}\}$ とする。 G に対応するライン表現の一方の直線 L_1 上において、頂点に対応する線分の端点の順序が、 $x_n, x_{n-1}, \dots, x_2, x_1, y_{n'}, y_{n'-1}, \dots, y_2, y_1$ であるとする。

$wiper(x_i)$ とは、 L_1 上の点 p_1 と L_2 上の点 p_2 を結ぶ線分である。 p_1 は L_1 上の固定された点で、 x_i と $y_{n'}$ の間にある。 p_2 は L_2 上の点で、 x_{i+1} と点 q の間にある。ここで点 q とは、 $N(x_i) = N(x_{i+1})$ なら x_i 、もしくは $N(x_i) \setminus N(x_{i+1}) \neq \emptyset$ なら $N(x_i) \setminus N(x_{i+1})$ の中で最大の添字 j を持つ y_j である。例を示す。図 4.3 はチェーングラフ $G = (X, Y, E)$ である。図 4.4 は G のライン表現に $wiper(x_1)$ と $wiper(x_3)$ を示したものである。点 p_1 は x_1 と y_5 の間にある。 $wiper(x_1)$ では、 $N(x_1) \setminus N(x_2) \neq \emptyset$ なので、 q は $N(x_1) \setminus N(x_2)$ の中で最大の添字番号を持つ y_1 である。 $wiper(x_3)$ では、 $N(x_3) = N(x_4)$ なので、 q は x_3 である。

4.2.2 チェーングラフの wiper による区間表現

チェーングラフ G において $wiper(x_i)$ は $|X|(0 \leq i \leq |X|-1)$ 本考えられる。各 $wiper(x_i)$ に対して区間表現 H_i を作ることができる。

区間表現の作り方を説明する。区間表現の任意の区間は数直線上の整数を端点に持つ閉区間である。 $wiper(x_i)$ は $[0, 0]$ を通る垂直線とする(ただし $wiper(x_i)$ は区間表現に含まれない。) L_2 上で点 q より右に端点をもつ線分と左に端点を持つ線分に分けて考える。 L_2

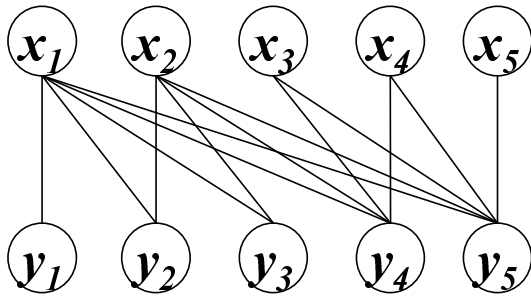


図 4.3: チェーングラフ G

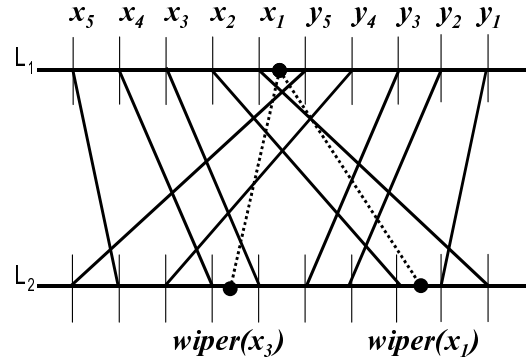


図 4.4: 図 4.3 のライン表現と wiper の例

上で点 q より右に端点をもつ線分 x_i は区間 $[0, (d(x_i) - d(x_i))]$, 線分 y_j は点 $i - j + 1$ に配置する. L_2 上で点 q より左に端点をもつ線分 x_i は点 $i - i$, 線分 y_j は区間 $[i - (d(y_i), 0]$ に配置する. $wiper(x_1)$ の例を図 4.5 に, $wiper(x_3)$ の例を図 4.6 に示す.

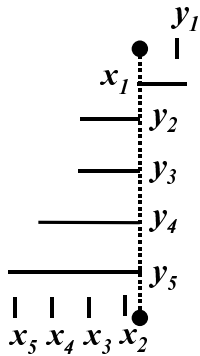


図 4.5: 図 4.4 の $wiper(x_1)$ による区間表現

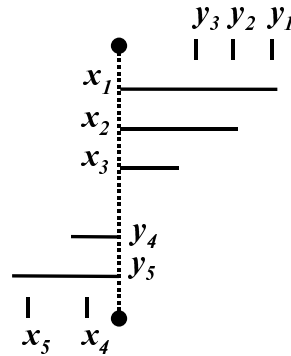


図 4.6: 図 4.4 の $wiper(x_3)$ による区間表現

4.2.3 チェーングラフ上のバンド幅問題を解くアルゴリズム

本節では, チェーングラフ上のバンド幅問題を解くアルゴリズムの概要について述べる. チェーングラフ上の最適なレイアウトは, チェーングラフの wiper によるすべての区間表現について minimum proper interval completion を解くことにより求まる.

まず, 前処理として $wiper(x_0)$ による区間表現 H_0 の最適なレイアウトを求める. H_0 は閾値グラフである. したがって定理 2 より $O(n)$ 時間で H_0 の最適なレイアウトを求めることができる. $wiper(x_0)$ の例を図 4.7 に, 区間表現 H_0 を図 4.8 に示す.

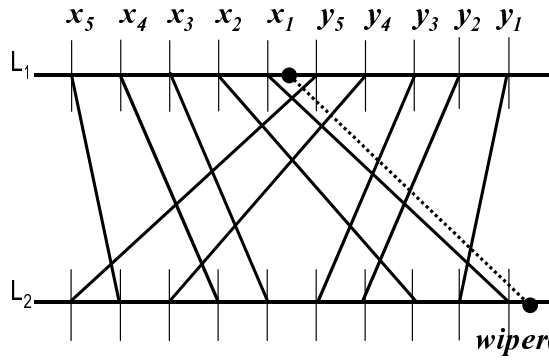


図 4.7: ライン表現の $wiper(x_0)$ の例

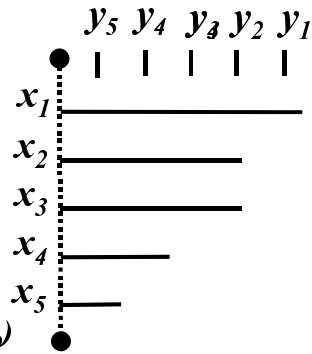


図 4.8: G の $wiper(x_0)$ による区間表現

次に, $wiper(x_1)$ により得られた区間表現 H_1 について考える. H_1 は 2 つの閾値グラフで表現されていることが分かる. つまり, L_2 上で点 q より右に端点をもつ線分に対応する区間表現と, 左に端点を持つ線分に対応する区間表現が, それぞれ閾値グラフとなる. 図 4.5 を例に示すと, $\{x_2, x_3, x_4, x_5, y_2, y_3, y_4, y_5\}$ により導出される閾値グラフと, $\{x_1, y_1\}$ により導出される閾値グラフである. この 2 つの閾値グラフに対して, 閾値グラフのバンド幅を求めるアルゴリズムと同様の計算をする. 結果, H_1 の頂点数に比例する時間で H_1 の minimum proper interval completion を求めることができる. この計算をすべての $wiper(x_i)$ について解くことで, 最適なレイアウトを得ることができる.

$G = (X, Y, E)$ を与えられたチェーングラフとする. H_i の minimum proper interval completion で得られた区間表現を H'_i とする. H'_i において, 頂点集合 X と Y は, ライン表現の一方の直線 L_2 上で点 q を境にして, 2 つに分割される. 頂点集合 X は, 点 q より左に端点をもつ線分 $x_{i'}$ は点 $i' - i$, 点 q より右に端点をもつ線分 $x_{i'}$ は区間 $[0, (d(x_{i'}) - d(x_i))]$ の 2 つに分割される. 頂点集合 Y は, 点 q より左に端点をもつ線分 $y_{j'}$ は区間 $[i - (d(y_{i'}), 0]$, 点 q より右に端点をもつ線分 $y_{j'}$ は点 $i - j + 1$ の 2 つに分割される. このとき, H'_i において, 頂点集合 X は $X = X_0 \cup X_1 \cup X_2$, 頂点集合 Y は $Y = Y_0 \cup Y_1 \cup Y_2$ に分割される. X_0 は H'_i において対応する区間の右端点が $i' - i$ である頂点の集合である. X_1 は H'_i において対応する区間の左端点が $i' - i$ である頂点の集合である. X_2 は H'_i において対応する区間が $[0, (d(x_{i'}) - d(x_i))]$ である頂点の集合である. Y_0 は H'_i において対応する区間が $[i - (d(y_{i'}), 0]$ である頂点の集合である. Y_1 は H'_i において対応する区間の右端点が点 $i - j + 1$ である頂点の集合である. Y_2 は H'_i において対応する区間の左端点が点 $i - j + 1$ である頂点の集合である. ただし, $X_0, X_1, X_2, Y_0, Y_1, Y_2$ のどれかは, 空である可能性もある. このとき, クリークを作る頂点集合の候補が以下の 3 つに絞られる. 候補は, (1) $\{X_0\} \cup \{Y_0\}$, (2) $\{X_1 \cup Y_0 \cup X_2 \cup Y_1\}$, (3) $\{X_2 \cup Y_2\}$ のどれかである. (1) により作られるクリークを LC , (2) により作られるクリークを CC , (3) により作られるクリークを RC とする. これから先のアルゴリズムでは, この 3 つのクリークの大きさを計算する. 一般性を失わないように, H_i から H_{i+1} へ更新するときの計算を考える. このとき, L_2 上

で $wiper(i)$ と $wiper(i+1)$ の間に端点を持つ頂点についてのみ考える． x_i は LC か CC のどちらかに属している． x_i が LC に属していれば， LC の大きさが 1 小さくなり， CC に入っていれば CC は変化なしである． L_2 上で $wiper(i)$ と $wiper(i+1)$ の間に端点を持つ頂点集合を， Y_{i+1} とする． Y_{i+1} は CC または RC に属する．そして， Y に属する頂点で一度 RC に属した頂点は再度 CC に属することはない．したがって，すべての頂点は高々定数回操作される． H_0 の計算に $O(n)$ 時間，すべての H_i の計算に $O(n)$ 時間かかる．したがって，チェーングラフ上のバンド幅問題を解くアルゴリズム全体の計算時間は $O(n)$ 時間である．チェーングラフの minimum proper interval completion の例を図 4.9 に示す． $X_0 = \{x_1, x_2, \dots, x_i\}$ ， $X_1 = \{x_{i+1}, x_{i+2}, \dots, x_j, \dots, x_p\}$ ， $X_2 = \{x_{p+1}, \dots, x_q, \dots, x_n\}$ である． $Y_0 = \{y_1, y_2, \dots, y_a, \dots, y_{b-1}\}$ ， $Y_1 = \{y_b, \dots, y_c, \dots, y_d\}$ ， $Y_2 = \{y_{d+1}, \dots, y_{n'}\}$ である．

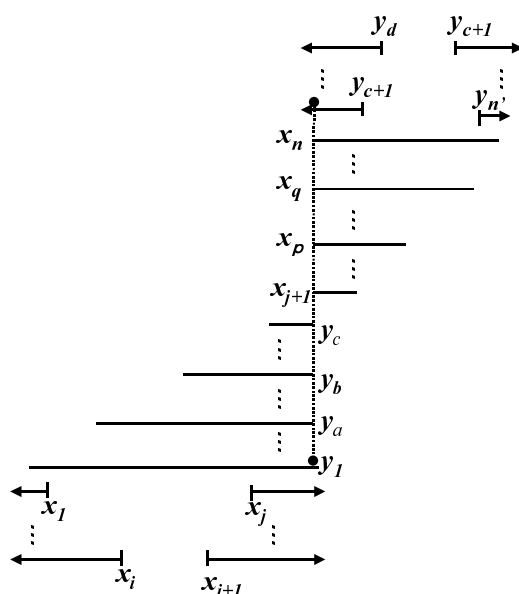


図 4.9: チェーングラフの minimum proper interval completion の例

4.2.4 チェーングラフの最適なレイアウトの性質

以上の議論により，以下の補題が得られる．

補題 5. チェーングラフ $G = (X, Y, E)$ ，頂点集合 $X = \{x_1, x_2, \dots, x_n\}$ ， $Y = \{y_1, y_2, \dots, y_{n'}\}$ は隣接点集合の包含関係でソートされているとする．このとき，頂点集合 X は X_1, X_2, X_3 ， Y は Y_1, Y_2, Y_3 に分割され， $X_1 < Y_1 < X_2 \cup Y_2 < X_3 < Y_3$ の順に並ぶ．ただし，ある $1 \leq i \leq j \leq p \leq q \leq n$ について， $X_1 = \{x_1, x_2, \dots, x_i\}$ ， $X_2 = \{x_{i+1}, x_{i+2}, \dots, x_j, \dots, x_p\}$ ， $X_3 = \{x_{p+1}, \dots, x_q, \dots, x_n\}$ であり，ある $1 \leq a \leq b \leq c \leq d \leq n'$ について， $Y_1 =$

$\{y_1, y_2, \dots, y_a, \dots, y_{b-1}\}$, $Y_2 = \{y_b, \dots, y_c, \dots, y_d\}$, $Y_3 = \{y_{d+1}, \dots, y_{n'}\}$ である. $X_2 \cup Y_2$ はどんな配置でもよい.

4.3 2部パーミュテーショングラフ上の k バンド幅問題

本節では, 2部パーミュテーショングラフ上の k バンド幅問題を解くアルゴリズムについて述べる.

4.3.1 2部パーミュテーショングラフの最適なレイアウトの持つ性質

補題 6. [11] $G = (X, Y, E)$ を 2部パーミュテーショングラフとし, G のバンド幅は k 以下とする. また, L_1, L_2 を G のライン表現の 2 本の平行線とする. L_1 上に左から現れる X の頂点に対応する線分の端点の順序を $P_x = (x_1, \dots, x_n)$, Y の頂点に対応する線分の端点の順序を $P_y = (y_1, \dots, y_{n'})$ とする. このとき, バンド幅が k 以下で, X の頂点は P_x の順で, Y の頂点は P_y の順で現れるレイアウトが存在する.

補題 2, 5, 6 より以下の補題が得られる.

補題 7. 2部パーミュテーショングラフ $G = (X, Y, E)$, 頂点集合 $X = \{x_1, x_2, \dots, x_n\}$, $Y = \{y_1, y_2, \dots, y_{n'}\}$ は隣接点集合の包含関係でソートされているとする. このとき, 以下を満たす最適なレイアウトが存在する.

1. 頂点集合 V_0, V_m は, $V_i^1, V_i^2, V_i^3 (i = 0, m)$ の高々 3 つの頂点集合に分割される.
2. 頂点集合 $V_i (1 < i < m)$ は高々 5 つの頂点集合 $V_i^j (j = 0, 1, \dots, 4)$ に分割される.
3. (1) と (2) において, 頂点集合の順序は維持される. つまり, V_i^j の各頂点は V_i^{j+1} より左に配置される.
4. 頂点集合 V_i において, レイアウトの順序は維持される.
5. レイアウトは以下の頂点集合の順序を満たす.

$$V_0^0 < V_1^0 < V_0^1 \cup V_1^1 < V_0^2 < V_1^2 < V_2^0 < V_1^3 \cup V_2^1 < V_1^4 < V_2^2 < V_3^0 < V_2^3 \cup V_3^1 < V_2^4 < V_3^2 < V_4^0 < V_3^3 \cup V_4^1 < V_3^4 < \dots < V_{m-1}^0 < V_{m-2}^3 \cup V_{m-1}^1 < V_{m-2}^4 < V_{m-1}^2 < V_m^0 < V_{m-1}^3 \cup V_m^1 < V_{m-1}^4 < V_m^2$$

この頂点集合の順序は, $1 \leq i \leq m-1$ について,

$$V_i^0 < V_{i-1}^3 \cup V_i^1 < V_{i-1}^4 < V_i^2 < V_{i+1}^0 < V_i^3 \cup V_{i+1}^1 < V_i^4$$

から成る.

4.3.2 アルゴリズム

まず、アルゴリズムの概要について説明する。このアルゴリズムは次の2つのステップに分かれる。まず、2部パーミュテーショングラフ G をチェーングラフの列に分解する。次に、チェーングラフのバンド幅 k 以下のレイアウトを求め、これを次のチェーングラフとバンド幅 k 以下になるように上手く配置していく。この上手く配置していくという所がとても難しい。しかし、頂点集合の並ぶ順番が補題7により分かった。よって、後者のステップを効率的に計算することができるようになった。

Algorithm 1: 提案アルゴリズム

Input: 2部パーミュテーショングラフ $G = (X, Y, E)$, G のライン表現上における順序 $X = \{x_1, x_2, \dots, x_n\}$ と $Y = \{y_1, y_2, \dots, y_{n'}\}$, 自然数 k

Output: もし存在するならバンド幅 k 以下のレイアウト。そうでなければ “no”。

頂点集合 $X \cup Y$ に対して, $V_0 = \{x_0\}, V_1, V_2, \dots, V_m$ を計算

for $j = 1, 2, \dots, m - 1$ do

 チェーングラフ $G_j = (V_j, V_{j+1}, E_j)$ を作る

 for $i = 0, 1, \dots, |V_j|$ do

 区間グラフ H_i^j を作る

 for $i' = 0, 1, \dots, |V_{j-1}|$ do

LC_i の大きさが k になるまで CC_i から頂点を移動

CC_i の大きさが k になるまで CC_i から RC_i に頂点を移動

 for $i'' = 0, 1, \dots, |V_{j+1}|$ do

 if V_j と V_{j+1} 間の2頂点間でバンド幅 k を超えるものが存在する then

V_{j+1} に隣接する V_j の頂点を左から右に移動

頂点集合 $X \cup Y$ に対して, $V_0 = \{y_0\}, V_1, V_2, \dots, V_m$ を計算し, 上記のループを再度計算

ある i に対して, $bw(H_i^j) \leq k$ が存在しなければ “no” を返す

4.3.3 アルゴリズムの説明

$G = (X, Y, E)$ を 2 部パーミュテーショングラフとする． G のライン表現の一方の直線上において，最初に出現する端点を持つ線分に対応する頂点で， X に属する頂点を x_0 ， Y に属する頂点を y_0 とする．

x_0 を先頭とするととき， G がバンド幅 k 以下のレイアウトを持つかどうか計算する．そのようなレイアウトが存在しないとき， y_0 を先頭として， G がバンド幅 k 以下のレイアウトを持つかどうか計算する．

x_0 を先頭とするととき，前処理として，頂点集合 $X \cup Y$ に対して， $V_0 = \{x_0\}, V_1, V_2, \dots, V_m$ を計算する．これは補題 2 よりライン表現から計算できる．最初のループでは，チェーングラフの列 G_1, G_2, \dots, G_m を作る．一般性を失わないように，チェーングラフ $G_i = (V_i, V_{i+1}, E_i)$ の計算をするときを説明する．まず， G_i の区間表現 $H_i (i = 0, 1, \dots, |V_i|)$ から， G_i のバンド幅 k 以下のレイアウト σ_i を求める．補題 7 より， G_i の計算をするときは， $V_i^0 < V_{i-1}^3 \cup V_i^1 < V_{i-1}^4 < V_i^2 < V_{i+1}^0 < V_i^3 \cup V_{i+1}^1 < V_i^4 < V_{i+1}^2$ の頂点集合に，どのように頂点が配置されるかを計算すればよいことが分かる．ここで， σ_i の計算方法について説明する．まず， $V_i^2 < V_{i+1}^0 < V_i^3 \cup V_{i+1}^1 < V_i^4 < V_{i+1}^2$ に， G_i の頂点集合に配置する．これは，チェーングラフ上のバンド幅問題を解くアルゴリズムより得られるレイアウトの順に頂点を配置することである．ただし， V_i^0, V_i^1 にすでに配置されている頂点は，ここでは配置しない．次に，いま配置した頂点をレイアウト $V_i^2 < V_{i+1}^0 < V_i^3 \cup V_{i+1}^1 < V_i^4 < V_{i+1}^2$ がバンド幅 k 以下になるように配置する方法を説明する． V_i^3 の頂点を V_i^2 に LC_i が k になるまで頂点を移動する．この時の移動方法は， V_i^3 の左端の頂点を V_i^2 の右端に移動することである．そして， CC_i の大きさが k になるまで V_{i+1}^1 から V_{i+1}^2 に頂点を移動する（はじめから $CC_i \leq k$ のときはなにもしない．）この時の移動方法は， V_{i+1}^1 の右端の頂点を V_{i+1}^2 の左端に移動することである．次に， G_i と G_{i-1} のバンド幅 k 以下のレイアウトを持つ頂点の配置を計算する．これは頂点集合 V_{i+1} と V_i の隣接する 2 頂点がバンド幅 k 以下になるように計算することである．具体的には， $V_i^0 \cup V_i^1$ と $V_{i+1}^0 \cup V_{i+1}^1 \cup V_{i+1}^2$ との隣接する 2 頂点がバンド幅 k 以下になるように V_i に属する頂点を再配置することである．この計算を G_1 から G_m まで適用して，アルゴリズムがバンド幅 k 以下のレイアウトを求めることができるのであれば，そのレイアウトを出力する．

y_0 を先頭としたとき， $V_0 = \{y_0\}, V_1, V_2, \dots, V_m$ とを求め，得られたチェーングラフの列に対して x_0 を先頭としたときと同じアルゴリズムを適用する．こちらでもバンド幅 k 以下のレイアウトが得られなかった場合は，アルゴリズムは “no” を出力する．

4.3.4 データ構造

ここでは，2 部パーミュテーショングラフ上の k バンド幅問題を解くために用いるデータ構造について説明する．

補題 7 の任意の頂点集合 $V_i^j (0 \leq i \leq m, 0 \leq j \leq 4)$ は以下の情報を持つ．

1. V_i^j は, V_i^j に属する頂点集合を双方向リストで持つ .
2. V_i^j に属する各頂点は, 隣接点集合の中で最も添字番号の大きい頂点と小さい頂点の情報を持つ .
3. V_i^j は, $|V_i^j|$ を持つ .
4. V_i^j は, 補題 7 の (5) のレイアウトにおける V_i^j の次に出現する頂点集合へのリンクを持つ .

4.3.5 計算時間

ここでは, 4.3.4 のデータ構造を用いると, 2部パーミュテーショングラフ上の k バンド幅問題を線形時間で解くことができることを示す .

$G = (X, Y, E)$ を 2部パーミュテーショングラフとする . まず, 前処理として, 頂点集合 $X \cup Y$ に対して, $V_0, V_1, V_2, \dots, V_m$ を計算するときを考える . この計算はライン表現より計算できる . このとき, どの線分も定数回しか調べないので, $O(n)$ 時間である . このときに, 同時に各頂点は隣接する頂点の中で, 最も添字番号の大きい頂点と小さい頂点の情報を得ることができる .

一般性を失わないように, チェーングラフ $G_i = (V_i, V_{i+1}, E_i)$ の計算をするときを考える . G_i のバンド幅 k 以下を満たすレイアウトを $\sigma_i (0 \leq i \leq |V_i|)$ とする . このレイアウトの計算はチェーングラフ上のバンド幅問題を解くアルゴリズム同様に計算できる . したがって, 計算時間は $O(n)$ 時間である .

$V_i^0 \cup V_i^1$ と $V_{i+1}^0 \cup V_{i+1}^1 \cup V_{i+1}^2$ との隣接する 2 頂点がバンド幅 k 以下になるように計算するときを考える . このとき, V_{i+1} に属する各頂点を左から順に最も離れている隣接する頂点との距離が k 以下であるかどうかを調べる . この距離が k より大きければ, 頂点集合の再配置を行う . この計算は, V_i^0 と V_i^1 に属する頂点を左から順に調べることでできる . 距離の計算は定数時間でできる . いま調べている V_i^0 または V_i^1 に属するある頂点が, V_{i+1} に属するある頂点の最も離れている頂点とする . 補題 7 より, この 2 頂点間には高々定数個の頂点集合しかない . 4.3.4 のデータ構造より, 各頂点集合は自分の大きさと自分のすぐ右に来る頂点集合の情報を持っている . よって, この 2 頂点間の距離は, 定数個の頂点集合間のリンクを辿ることで距離を計算できる . また, V_{i+1} に属するある頂点に対して, 距離が k 以下であると分かった頂点についてはレイアウトが確定する . そして, レイアウトが確定した頂点は再度調べられることがない . 距離が k より大きい頂点が見つかったときは, その頂点とその頂点より右側にある頂点すべてが, 補題 7 の (5) の頂点集合の順序における V_i に属する次の頂点集合の左側に再配置される . 各頂点集合は頂点の情報を双方向リストで持っているので, 頂点集合の再配置は定数時間で行うことができる . また補題 7 より, 各頂点集合 $V_i (0 \leq i \leq m)$ は高々 5 つに分割されることが分かる . よって, 再配置は高々定数回しか行われぬ . したがって, V_{i+1} に属する各頂点に対して, 隣接する

2 頂点がバンド幅 k 以下になるように計算するために必要な計算時間は定数時間である．これらの議論により， $V_i^0 \cup V_i^1$ と $V_{i+1}^0 \cup V_{i+1}^1 \cup V_{i+1}^2$ との隣接する 2 頂点がバンド幅 k 以下になるように計算するために必要な計算時間は $O(n)$ 時間である．

x_0 を先頭とするときの計算と， y_0 を先頭とするときの計算は同様にできるため，同じ計算時間である．

以上の議論により，この提案アルゴリズム全体の計算時間は線形時間である．

第5章 おわりに

本稿では，2部パーミュテーショングラフ上の k バンド幅問題を線形時間で解くことができることを示した．

今後の課題としては，一般のグラフに対してのバンド幅問題を解く厳密なアルゴリズムの計算時間の改善や，パーミュテーショングラフ上の効率の良いアルゴリズムの開発などがある．

謝辞

本研究を行うにあたり，日頃より懇切丁寧な指導を賜りました上原隆平准教授に心より感謝いたします。また，浅野哲夫教授，清見礼助教には，ゼミなどにおいて適切な御教示を頂き，厚く御礼申し上げます。浅野研究室，上原研究室の学生の皆様にも公私にわたり，様々な場面でお世話になりました。この場を借りて感謝いたします。最後に，大学院での研究を支えてくれた家族に感謝いたします。

参考文献

- [1] Y.L. Lai and K. Williams. A survey of solved problems and applications on bandwidth, edgesum, and profile of graphs. *J. of Graph Theory*, 31(2):75–94, 1999.
- [2] H. Kaplan, R. Shamir, and R.E. Tarjan. Tractability of Parameterized Completion Problems on Chordal, Strongly Chordal, and Proper Interval Graphs. *SIAM J. on Comp.*, 28(5):1906–1922, 1999.
- [3] C.H. Papadimitriou. The NP-Completeness of the bandwidth minimization problem. *Computing*, 16:263-270, 1976.
- [4] B. Monien. The Bandwidth Minimization Problem for Caterpillars with Hair Length 3 is NP-Complete. *SIAM J. Alg. Disc. Meth.*, 7(4):505–512, 1986.
- [5] M. Cygan and M. Pilipczuk. Faster Exact Bandwidth. In *WG 2008*, LNCS, Springer-Verlag, 2008.
- [6] R. Uehara. Bandwidth of Bipartite Permutation Graphs. In *ISAAC 2008*, pp. 824–835. LNCS Vol. 5369, Springer-Verlag, 2008.
- [7] A.P. Sprague. An $O(n \log n)$ Algorithm for Bandwidth of Interval Graphs. *SIAM J. on Disc. Math.*, 7:213–220, 1994.
- [8] R. Uehara and Y. Uno. On Computing Longest Paths in Small Graph Classes. *International Journal of Foundations of Computer Science*, 18(5):911-930,2007.
- [9] X. Deng, P. Hell, and J. Huang. Linear-time Representation Algorithms for Proper Circular-arc Graphs and Proper Interval Graphs. *SIAM Journal on Computing*, 25(2):390-403, 1996.
- [10] H. Kaplan and R. Shamir. Pathwidth, Bandwidth, and Completion Problems to Proper Interval Graphs with Small Cliques. *SIAM Journal on Computing*, 25(3):540–561, 1996.

- [11] P. Heggernes, D. Kratsch, and D. Meister. Bandwidth of bipartite permutation graphs in polynomial time. In *Latin American Theoretical Informations Symposium(LATIN)*, pages 216-227. Lecture Notes in Computer Science Vol.4957, Springer-Verlag, 2008.