

平成22年度 博士論文

de Bruijn 族のグラフの分解及び
情報散布への応用に関する研究

群馬大学大学院工学研究科電子情報工学領域

津野 崇寛

指導教員 荒木 徹 准教授

平成23年3月

目次

第 1 章	序論	1
第 2 章	諸定義	6
2.1	グラフとダイグラフ	6
第 3 章	de Bruijn 族のグラフ	13
3.1	de Bruijn ダイグラフ, Kautz ダイグラフ	13
3.2	一般化 de Bruijn ダイグラフ, 一般化 Kautz ダイグラフ	16
第 4 章	cycle-rooted tree を用いた de Bruijn 族グラフの分解	18
4.1	de Bruijn ダイグラフの cycle-rooted tree による分解	18
4.2	Kautz ダイグラフの分解	25
4.2.1	ダイグラフの coloring	25
4.2.2	2-cycle-rooted tree を用いた同型因子分解	26
4.2.3	cycle-rooted tree による分解の一般化	27
4.3	一般化 de Bruijn および一般化 Kautz ダイグラフの分解	34
4.3.1	loop-rooted tree による因子分解	36
4.3.2	loop-rooted tree による同型因子分解	39
4.3.3	複数のループを同時に含む同型因子分解	44
第 5 章	de Bruijn ダイグラフ間の包含関係	48
5.1	準同型写像を用いた de Bruijn ダイグラフの分解	48
5.1.1	m -反復準同型写像によるダイグラフ	49
5.2	一般化 de Bruijn ダイグラフの包含関係	53
5.2.1	ループを用いた同型因子分解のための必要条件	53
5.2.2	一般化 de Bruijn ダイグラフの積グラフの分解	57
第 6 章	グラフ上での情報散布問題	60
6.1	ネットワーク上での情報散布問題	60
6.2	情報散布問題の設定	61
第 7 章	de Bruijn 族グラフ上でのマルチソースブロードキャスト	64
7.1	de Bruijn ダイグラフ上でのマルチソースブロードキャスト	64
7.1.1	全域 cycle-rooted tree を用いたアルゴリズム	65

7.1.2	cycle-rooted tree による分解を用いたアルゴリズム	65
7.2	Kautz ダイグラフ上でのマルチソースブロードキャスト	68
7.2.1	大きな root-cycle を持つ全域 cycle-rooted tree の構成	69
7.2.2	全域 cycle-rooted tree を用いたアルゴリズム	77
7.2.3	SPCRT に用いられない弧の性質	83
7.2.4	cycle-rooted tree による分解を用いたアルゴリズム	88
7.3	各アルゴリズムに必要なラウンド数の比較	94
第 8 章	まとめと今後の課題	98
謝辞		101
参考文献		102
付 録 A	本論文で与えた定理の別証明	107

第1章 序論

情報通信技術の進歩により社会全体の情報化が進む近年，計算機に対して多くの要求がなされており，個々の目的に合う計算機の設計が必要となっている．特に複雑な計算処理の高速化は極めて大きな要求である．近年ではそれらの要求を満たすため，計算機やプロセッサの性能向上やメモリの大容量化などが著しく発展してきている．一方で，個々の計算機の性能向上だけでは計算処理全体の性能向上には限界があるため，複数の計算機を用いて高速な処理を行う並列計算機，分散処理システムの研究開発が進められている．並列計算機や分散処理システムの設計において，個々の計算機の性能だけでなく，プロセッサ間での相互通信を高速化することが全体の処理の高速化へとつながる．プロセッサ間の通信を高速化するため，相互結合網自体の改善およびその上で効率良く情報をやり取りする必要性が高まってきている．本研究ではこれらの要求に対し，グラフ理論を用いた二つのアプローチについて言及する．一つはグラフによって計算機相互結合網をモデル化し，その性質について考察する場合であり，もう一つはグラフアルゴリズムを考察する場合である．

並列計算機，分散処理システムを設計する上で，その性能を決定付ける大きな要因の一つとして各計算機やプロセッサをどのようにして接続させるかという相互結合網の選択が挙げられる．この相互結合網に対する性能についての問題を扱う際に相互結合網のプロセッサや計算機などの計算ノードを頂点，計算ノード間の通信リンクを辺に置き換えることで相互結合網をグラフおよびダイグラフでモデル化して考えることができる．

これまでの研究で相互結合網のモデルとして様々なグラフのクラスが提案されている．そのようなネットワークポロジのためにグラフとして提案されているものにパスやサイクル等の単純な構造を持つグラフや，それらを組み合わせて構成されるグリッドやトーラス，複雑な定義を持つキューブ族グラフ，バタフライ，de Bruijn ダイグラフ，Kautz ダイグラフなどが挙げられる．グラフのクラスはそれぞれの構成法を持ち，その構成法に応じた研究が存在する．ハイパーキューブなどの再帰的な構成法を持つグラフは一様性が高く，相互結合網としてのよい性質を持つことが多い．一方で，グラフの代数的な構成法は一般的に対称性が高く，拡張が容易である．これらのモデル化したグラフおよびダイグラフの性質を考察することにより実際の相互結合網の構造的な性質を解析することが可能となる．代数学を用いて表現される代表的なグラフクラスとして Cayley グラフがある．その中で，de Bruijn ダイグラフや Kautz ダイグラフは文字列を用いた再帰的な構成法と代数学による構成法を持ち，ライン(ダイ)グラフを用いた演算によって表現することも可能であるため，複数の観点からその構造を解析することができる．

相互結合網の性能を測るグラフ的な性質として、直径や次数、一様性や耐故障性、ルーティングアルゴリズムなどが挙げられる。直径が小さければ通信遅延は少なくて済み、次数を小さく抑えることは費用の削減につながる。構造に一様性があれば通信方法は均一化でき、耐故障性は高く、ルーティングアルゴリズムは容易であることが望ましい。しかしながら、これらの中にはトレードオフの関係にあるものも存在し、全ての要求を満たすようなトポロジーは存在しない。例えば、与えられた頂点数のもとで直径と次数を同時に小さくすることは不可能であり、この問題の限界は Moore bound として与えられている。そのため、用途に応じて適切な相互結合網を選択する必要がある。de Bruijn 族のグラフおよびダイグラフはその中でも多くの要求を満たしており、連結性や耐故障性に優れ、前述の Moore bound に対しても近い値を持つ。そのため、相互結合網として多くの研究が行なわれている。

グラフの埋め込み問題は理論的にも応用的にも広く研究されている。理論的な問題として、グラフを他のグラフへ埋め込む問題として与えられており、グラフ間の包含関係を示すものとして捉えられる。特に、埋め込むグラフが複数である場合にはそれぞれが互いに辺を共有しないことを要求され、同じグラフを複数埋め込む特別な場合としてグラフの分解問題が多く研究されている。分解するグラフがお互いに同型であること、あるいは埋め込み先のグラフと頂点数が等しい(全域である)ことは埋め込み先のグラフの一様性や耐故障性を保証し得る。そのため、グラフの同型分解、同型因子分解による結果は埋め込み可能性や情報散布の問題に応用される可能性が高い。

並列計算機を用いて計算処理の高速化を実現するために、適した相互結合網を選択するという一方で、各計算機間でどのように情報のやり取りを行なうかということも重要な問題として扱われる。並列計算機上では個々の計算機がお互いに情報を送受信しながら処理を行なうため、情報交換の効率を上げることはシステム全体の処理速度の上昇につながる。このような要求に対し、相互結合網としてモデル化したグラフ上で効率よく情報交換を行うためのグラフアルゴリズムを提案することが重要となる。情報交換に関する問題に対して様々な研究が行なわれている。特に、ある計算機が保持している情報を他の計算機に送信する情報散布は基本的かつ重要な問題であり、通信のためのスキームに関する研究が多くなされている。ブロードキャストとは、ある一つのプロセッサが持つ情報をネットワーク上の他のすべてのプロセッサに送信するための通信スキーム (one-to-all communication scheme) である。一方で、ネットワーク上のすべてのプロセッサがお互いに異なる情報を持ち、すべてのプロセッサがすべての情報を得るための通信スキーム (all-to-all communication scheme) はゴッシングと呼ばれる。これらの通信スキームは基本的な情報散布問題として知られており、様々なグラフクラスに対する研究が行われている [31]。

情報散布問題の目的は与えられた制約のもとで情報のやり取りにかかる時間を最小化することである。しかしながら、実際の情報散布においてはネットワーク上のプロセッサやリンク等の要素が持つ特性による制約が存在する。これらの制約のもとで、ネットワーク上の(全てではない)複数のプロセッサが散布すべき別個の情報を持つ場合を考えると、これらの

アルゴリズムの応用では冗長な部分が多くなり、余り効率的ではないことが起こり得る。この冗長性はブロードキャストイングにおいては一つの情報のみを取り扱っていることから、ゴシックングにおいてはすべての頂点が情報を持つと仮定していることから起こり得るものであり、これを改善するためにはある一定数の情報を同時に扱うことが可能な通信スキームが必要である。この考えから、上記の二つの情報散布問題の中間的な性質を持つ問題として、いくつかの計算機が個々に持つ情報を相互結合網上のすべての計算機へ送信する、マルチソースブロードキャストイングと呼ばれるスキーム (some-to-all communication scheme) が [41] で導入されている。

本研究で用いる制約のもとで、ネットワークは同期を取る、すなわち、ネットワーク上の各プロセッサは同時に通信を行なうことが可能である。ある一つのプロセッサから別のプロセッサへ、通信リンクを通じて一つの情報を送信するために必要な時間はラウンドと呼ばれる。ネットワーク上の通信を扱うモデルとして同時送受信モデルが提案されており [1], その制約は以下のように要約される。 [1] で導入されている send/receive model では以下のような制約を持つ。

- 各プロセッサはラウンド毎に高々一つの情報を受信できる。
- 各プロセッサはラウンド毎に高々一つの情報を送信できる。
- 各プロセッサはラウンド毎に異なる通信リンクを用いてある情報の受信と別の情報の送信を同時に行うことができる。

本論文でもこれらの制約のもとで情報散布のためのアルゴリズムを与える。

第2章では本論文中で用いられる基本的な用語および、用いられるグラフに関する定義を述べる。

第3章では de Bruijn 族グラフを紹介し、各ダイグラフの定義やそれらが持つ性質について説明する。

第4章では cycle-rooted tree を用いた de Bruijn 族の分解に関する考察を行う。cycle-rooted tree とは [27, 30] で提案されたグラフであり、ラインダイグラフに対してよい性質を持つことが知られている。また、cycle-rooted tree 上でブロードキャストイングを効率よく行うことができ、第6章で実際に cycle-rooted tree による分解を情報散布へ応用する。4.1 節では de Bruijn グラフについて cycle-rooted tree を用いた分解を与えている。cycle-rooted tree の根となる有向サイクルがループであるような loop-rooted tree と呼ばれるダイグラフによる de Bruijn ダイグラフの同型因子分解を与え、それをもとに分解する cycle-rooted tree の根となる有向サイクルの長さに応じて de Bruijn ダイグラフを分解するための方法を与える。4.2 節では Kautz ダイグラフに対し同様の分解を考え、4.2.1 節では根となる有向サイクルの長さが2である 2-cycle-rooted tree による同型因子分解を与え、4.2.2 節では cycle-rooted tree の有向サイクルの長さを一般化した場合の分解について述べる。また、4.3 節では、de Bruijn および Kautz ダイグラフを任意の頂点数に拡張した、一般化 de Bruijn ダイグラフ、

一般化 Kautz ダイグラフに対して loop-rooted tree をもととするようなグラフの分解について考察を行った。

第5章では de Bruijn 族のダイグラフに対し、グラフ間の包含関係について述べる。5.1 節では de Bruijn ダイグラフに対し再帰的に準同型写像を与えることで完全対称ダイグラフへの準同型写像を構成し、それをもとにして de Bruijn ダイグラフの有向サイクルによる分解を与えた。5.2 節では一般化 de Bruijn ダイグラフのクロネッカー積がその対応する一般化 de Bruijn ダイグラフにより同型因子分解されるための条件について考察している。de Bruijn 族とラインダイグラフ演算、クロネッカー積についての関係として de Bruijn ダイグラフおよび2進一般化 de Bruijn ダイグラフによる結果が示されており、これらのさらに一般的な結果について言及している。

第6章ではグラフ上での情報散布問題について、用語の定義を行いいくつかの先行研究について述べる。

第7章では各ダイグラフ上でのマルチソースブロードキャストを行うためのアルゴリズムをいくつか与え、それらの性能について評価を行う。7.1 節では de Bruijn ダイグラフに対するマルチソースブロードキャストについて考察する。7.1.1 節では全域 cycle-rooted tree を用いた既存のアルゴリズムを紹介し、7.1.2 節では4章で述べた cycle-rooted tree の分解を応用したアルゴリズムを提案して各アルゴリズムの性能の評価を行う。7.2 章では Kautz ダイグラフに対して同じくマルチソースブロードキャストを行うためのアルゴリズムを与える。7.2.1 節で与えるアルゴリズムは7.1.1 節の結果を Kautz ダイグラフに適用したものであり、7.2.2 節でのアルゴリズムは4章での分解をもととしたものである。

第8章では、本論文で得られている結果についてまとめ、今後の課題について述べる。

各章と関連論文およびその他の論文の関係

4章:

- T. Tsuno, Y. Shibata, “Factorization of generalized de Bruijn and Kautz digraphs by loop-rooted trees,” The 7th Japan Conference on Computational Geometry and Graphs, 2009.
- 津野崇寛, 柴田幸夫, “一般化 de Bruijn 及び Kautz ダイグラフの loop-rooted tree による分解,” 電子情報通信学会コンピュテーション研究会 COMP2009-57, 2009.

7章:

- 津野崇寛, 柴田幸夫, “Kautz ダイグラフ上の cycle-rooted tree を用いたマルチソースブロードキャスト,” 情報処理学会アルゴリズム研究会 2007-AL-113-2, 2007.
- 津野崇寛, 柴田幸夫, “Kautz ダイグラフ上の cycle-rooted tree を用いたマルチソースブロードキャスト,” 電子情報通信学会論文誌 A, Vol. J91-A, No. 2, pp. 212-222, (2008).
- T. Tsuno, Y. Shibata, “Multisource broadcasting on de Bruijn and Kautz digraphs using isomorphic factorizations into cycle-rooted trees,” IEICE Trans. Fundamentals, Vol. E92-A, no. 8, pp. 1757-1763, (2009).
- T. Tsuno, Y. Shibata, “An algorithm for multi-source broadcasting on Kautz digraphs using 2-cycle rooted trees,” The 12th Korea-Japan Joint Workshop on Algorithm and Computation, 2009.
- T. Tsuno, Y. Shibata, “An algorithm for multi-source broadcasting on Kautz digraphs using 2-cycle rooted trees,” IEICE Trans. Fundamentals, Vol. E93-A, no. 10, pp. 1800-1805, (2010).

第2章 諸定義

この章では本論文内で用いられる用語の定義について述べる。

2.1 グラフとダイグラフ

グラフ, ダイグラフ グラフ G は頂点と呼ばれる空でない有限集合 $V(G)$ と辺と呼ばれる $V(G)$ の要素の非順序対の集合 $E(G)$ によって定義される。 $V(G)$ を頂点集合, $E(G)$ を辺集合と呼び, $|V(G)|$ をオーダー, $|E(G)|$ をサイズと呼ぶ。 $(u, v) \in E(G)$ が存在するとき, 頂点 u と v は隣接しているといい, (u, v) は u および v と接続しているという。

ダイグラフ D は頂点集合 $V(D)$ と, 弧と呼ばれる $V(D)$ の要素の順序対の集合 $A(D)$ によって定義される。 $A(D)$ を弧集合と呼び, グラフと同様に $|V(D)|$, $|A(D)|$ をそれぞれオーダー, サイズと呼ぶ。 $(u, v) \in A(D)$ が存在するとき, u は v へ隣接しているといい, v は u から隣接しているという。 また, (u, v) は u から接続し, v へ接続しているという。 弧 (u, v) に対し, u を始点, v を終点と呼ぶ。

ループ, 双方向辺 ダイグラフ D に対し, $(u, u) \in A(D)$ が存在するとき, (u, u) をループと呼ぶ。 また, $(u, v) \in A(D)$ かつ $(v, u) \in A(D)$ であるとき, $(u, v), (v, u)$ をそれぞれ双方向辺と呼ぶ。

次数, 正則 グラフ G の頂点 u に対し, u に接続している辺の本数を次数と呼び, $\deg_G u$ あるいは単に $\deg u$ と表す。 G の任意の頂点 v に対し, $\deg v = d$ であるとき, G は d 正則であるという。

ダイグラフ D の頂点 u に対し, u へ接続している弧の本数を入次数と呼び $\text{id}_D u$ あるいは $\text{id } u$ と表す。 また, u から接続している弧の本数を出次数と呼び $\text{od}_D u$ あるいは $\text{od } u$ と表す。 D の頂点 u が持つループについては, u へ接続している弧および u から接続している弧のどちらもあるとみなす。 D の任意の頂点 v に対し $\text{id } v = d$ であるとき D は d 入正則であるといい, 同様に $\text{od } v = d$ であるとき D は d 出正則であるという。 D が d 入正則かつ d 出正則である場合, D は d 正則であるという。

有向ウォーク, 有向パス, 有向サイクル ダイグラフ D に対し, $u, v \in V(D)$ とする。 D における u - v 有向ウォーク W とは $1 \leq i \leq k$ なるすべての i に対して $(u_{i-1}, u_i) \in A(D)$ を

満たすような u と v を結ぶ頂点の列

$$W : u = u_0, u_1, \dots, u_k = v,$$

であり, 値 k を W の長さという. また, 列に含まれる頂点がすべて異なる有向ウォークを有向パスと呼ぶ. 有向サイクルとは, $k \geq 1$ を満たす $u-u$ 有向パスである. グラフ G に対し, オーダ $|V(G)|$ と等しい長さを持つ有向サイクルをハミルトン有向サイクルと呼ぶ.

セミパス ダイグラフ D の $u-v$ セミパスとは $1 \leq i \leq k$ なるすべての i に対して $(u_{i-1}, u_i) \in A(D)$ または $(u_i, u_{i-1}) \in A(D)$ を満たすような u と v を結ぶ頂点の列

$$u = u_0, u_1, \dots, u_k = v,$$

である. このとき, 列に現れる頂点は全て異なる頂点である.

部分ダイグラフ, 因子 二つのダイグラフ D, D_1 に対し, $V(D_1) \subseteq V(D)$ かつ $A(D_1) \subseteq A(D)$ であるとき D_1 は D の部分ダイグラフであるという. 特に, $V(D_1) = V(D)$ であるような場合, D_1 は D の全域部分ダイグラフあるいは因子であるという.

誘導 ダイグラフ D に対し, $U \subseteq V(D), U \neq \emptyset$ なる U が与えられたとき, 頂点集合として U を持ち, 弧集合として U の要素である頂点に接続し, また U の要素である頂点から接続されているような D のすべての弧を含むダイグラフを U による D の誘導部分ダイグラフと呼び, $\langle U \rangle$ と表す. 同様に, $X \subseteq A(D), X \neq \emptyset$ なる X に対し, 頂点集合として X に含まれる弧へ接続している D の頂点および X に含まれる弧から接続している頂点を持ち, 弧集合として X を持つようなダイグラフを X による弧誘導部分ダイグラフと呼び $\langle X \rangle$ と表す.

完全対称ダイグラフ, 完全ダイグラフ 完全対称ダイグラフとはすべての異なる二つの頂点 u, v に対し, 弧 (u, v) と (v, u) のどちらも含むようなダイグラフのことをいい, オーダ n の完全対称ダイグラフを K_n^* と表す. K_n^* のすべての頂点について, ループを一つ加えたダイグラフを完全ダイグラフと呼び, K_n^+ と表す.

底グラフ ダイグラフ D の底グラフとは, D のすべての弧から向きを取り除き, ループを削除し, 双方向辺を一つの辺と置き換えることで得られるグラフのことをいう.

連結, 成分 ダイグラフ D の任意の2頂点 u, v に対し, $u-v$ セミパスが存在するとき, D は連結あるいは弱連結であるといい, $u-v$ 有向パスが存在するとき, D は強連結であるという. ダイグラフ D の誘導部分ダイグラフの中で弱連結性に関して極大なものを D の成分と呼ぶ.

距離, 直径 ダイグラフ D の 2 頂点 u, v に対し, u から v への距離とは最も短い u, v 有向パスの長さのことをいい, $d(u, v)$ と表す. ダイグラフ D の直径とは $\max_{u, v \in V(D)} d(u, v)$ であり, $\text{diam } D$ と表す.

点素, 弧素 二つのダイグラフ D_1, D_2 に対し, $V(D_1) \cap V(D_2) = \emptyset$ であるとき, D_1 と D_2 は点素であるといい, $A(D_1) \cap A(D_2) = \emptyset$ であるとき D_1 と D_2 は弧素であるという.

同型, 自己同型 二つのダイグラフ D_1, D_2 に対し,

$$(u, v) \in A(D_1) \iff (\phi(u), \phi(v)) \in A(D_2),$$

を満たすような $V(D_1)$ から $V(D_2)$ への全単射 ϕ が存在するとき, D_1 と D_2 は同型であるといい, $D_1 \cong D_2$ と表す. このとき, ϕ を同型写像と呼ぶ. また, ダイグラフ D に対し, $V(D)$ から $V(D)$ への同型写像を自己同型写像と呼ぶ.

準同型, 頂点一様, 弧一様, 負荷 ダイグラフ G_1 がダイグラフ G_2 に準同型であるとは, 任意の $u, v \in V(G_1)$ に対し, $uv \in A(G_1)$ ならば $\phi(u)\phi(v) \in A(G_2)$ となるような $V(G_1)$ から $V(G_2)$ への全射 ϕ が存在することである. このとき, ϕ はダイグラフ V_1 から V_2 への準同型写像と呼ばれる.

ϕ をダイグラフ V_1 から V_2 への準同型写像とする. ϕ のもとでの頂点 $x \in V_2$ の負荷 $\text{load}_\phi(x)$ は x へ写像される V_1 の頂点の総数に等しい. 準同型写像 ϕ が頂点一様であるとは, V_2 の任意の頂点 x に対し,

$$\left\lfloor \frac{|V_1|}{|V_2|} \right\rfloor \leq \text{load}_\phi(x) \leq \left\lceil \frac{|V_1|}{|V_2|} \right\rceil \quad (2.1)$$

を満たすことである. また, 弧 $a = uv \in A(G_2)$ の負荷 $\text{load}_\phi(a)$ は $\phi(u') = u, \phi(v') = v$ かつ $u'v' \in A(G_1)$ となるような順序対 (u', v') の総数と等しい. 準同型写像 ϕ が弧一様であるとは, A_2 の任意の頂点 a に対し,

$$\left\lfloor \frac{|A_1|}{|A_2|} \right\rfloor \leq \text{load}_\phi(a) \leq \left\lceil \frac{|A_1|}{|A_2|} \right\rceil \quad (2.2)$$

を満たすことである.

ラインダイグラフ ダイグラフ D のラインダイグラフ $L(D)$ は以下で定義される.

$$V(L(D)) = A(D),$$

$$A(L(D)) = \{((u, v), (v, w)) \mid (u, v), (v, w) \in A(D)\}.$$

D から $L(D)$ を構成する操作をラインダイグラフ演算と呼ぶ. ダイグラフ D に対してラインダイグラフ演算を k 回施すことで得られるダイグラフを, ダイグラフ D の k 反復ラインダイグラフと呼び $L^k(G)$ と表す. 図 2.1 にラインダイグラフ演算の例を示す.

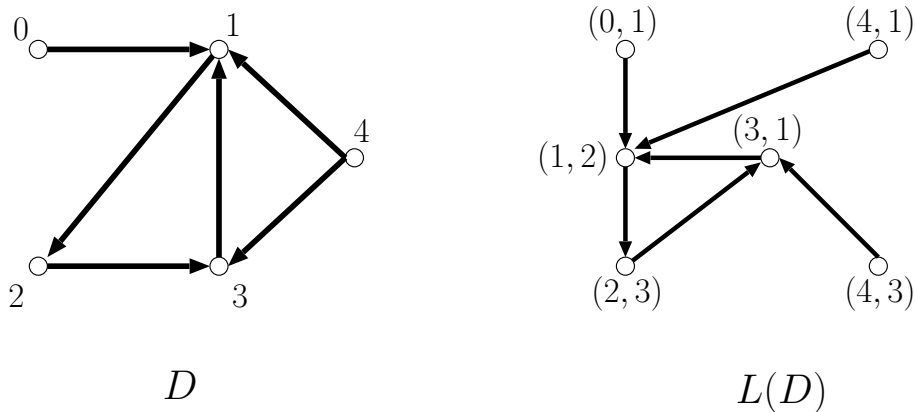


図 2.1: ラインダイグラフの例

クロネッカー積 ダイグラフ D_1 と D_2 のクロネッカー積 $D = D_1 \otimes D_2$ は頂点集合 $V(D) = V(D_1) \times V(D_2)$ を持ち, D 内の頂点 (u_1, u_2) が (v_1, v_2) に隣接するための必要十分条件が,

$$(u_1, v_1) \in A(D_1) \text{ and } (u_2, v_2) \in A(D_2)$$

となるようなダイグラフ D である.

分解, 因子分解, 同型因子分解 ダイグラフ D の弧集合 $A(D)$ に対し, $A_i \cap A_j = \emptyset, (i \neq j)$ かつ $\bigcup_{1 \leq i \leq k} A_i = A(D)$ を満たすような分割集合が存在するとき, D は $\langle A_1 \rangle, \langle A_2 \rangle, \dots, \langle A_k \rangle$ によって分解されるといい, $D = \langle A_1 \rangle \oplus \langle A_2 \rangle \oplus \dots \oplus \langle A_k \rangle$ と表す. 特に, 各 $\langle A_i \rangle$ がそれぞれ因子である場合に D は因子分解されるといい, かつそれぞれが同型である場合には D は同型因子分解されるという. ダイグラフ D が ダイグラフ H により同型因子分解されるとき, $D \mid H$ と表す.

有向木 ダイグラフ D が有向木であるとは, D が弱連結であり, 根と呼ばれる入次数が 0 の頂点の一つを持ち, 他の全ての頂点の入次数が 1 であるときをいう. 有向木の中で出次数が 0 の頂点は葉と呼ばれる. 定義より有向木は根から他の任意の頂点までの有向パスが一意に定まる. 根から頂点 v までの有向パスの長さを v の深さと呼び, 深さの最大値を有向木の高さと呼ぶ. 有向木の 2 頂点 u, v に対し, 弧 (u, v) が存在するとき, u は v の親であるといい, v は u の子であるという.

葉の深さが全て等しく, 葉以外の全ての頂点の出次数が d である有向木を完全有向 d 分木と呼び, 高さが h の完全有向 d 分木を $CT(d, h)$ で表す. $h \geq 1$ のとき, $CT(d, h)$ から $CT(d, h-1)$ の部分木を一つ削除することで得られる有向木を $CT^-(d, h)$ と表す. 図 2.2 に $CT(3, 2)$ および $CT^-(3, 2)$ を示す.

cycle-rooted tree

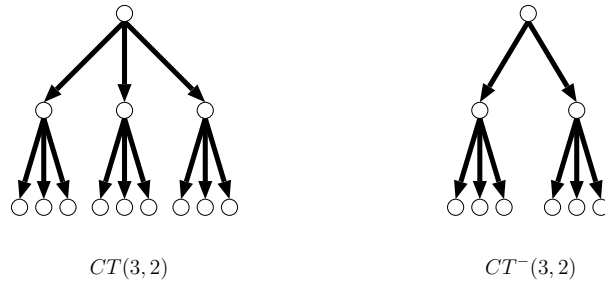


図 2.2: $CT(3, 2)$ および $CT^-(3, 2)$

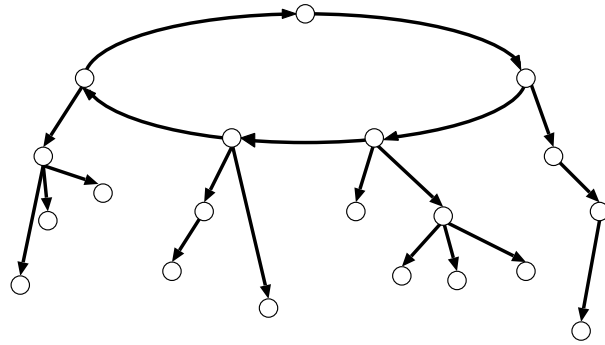


図 2.3: cycle-rooted tree の例

定義 2.1 (Hasunuma and Shibata [27, 30]) ダイグラフ D が cycle-rooted tree であるとは, D が root-cycle と呼ばれるただ一つの有向サイクルを持ち, 各頂点の入次数が全て 1 であるときをいう.

定義より, cycle-rooted tree はいくつかの有向木の根を有向サイクルとなるように弧で繋いだ形をしている. 特に, root-cycle がループ頂点であるような cycle-rooted tree を loop-rooted tree として表す. root-cycle 上の頂点をそれぞれ cycle-vertex と呼び, root-cycle を構成する弧を消去したときに得られる cycle-vertex を根とする有向木を collateral tree と呼ぶ. 各 collateral tree に含まれる頂点に対し, その根である cycle-vertex からの有向パスの長さをその頂点の深さとして定義し, すべての collateral tree の中で最大となる深さの値を cycle-rooted tree の高さとして定義する.

また, すべての cycle-vertex が collateral tree として $CT^-(d, h)$ を持つ cycle-rooted tree を高さ h の完全 d 進 cycle-rooted tree と表す. 図 2.3 に cycle-rooted tree の例を示す.

記号表

$a \mid b$	a は b を割り切る
$a \nmid b$	a は b を割り切らない
$a \perp b$	a と b は互いに素
$a \equiv b \pmod{m}$	a は m を法として n と合同
$a \bmod m$	a の m による剰余
$a \oplus_m b$	a と b の m を法とする和
$a \ominus_m b$	a と b の m を法とする差
$\gcd(a, b)$	a と b の最大公約数
$\text{lcm}(a, b)$	a と b の最小公倍数
$\lfloor r \rfloor$	r 以下である最大の整数
$\lceil r \rceil$	r 以上である最小の整数
\mathbb{Z}_n	$\{0, 1, \dots, n-1\}$
μ	メビウス関数 : $\mu(d) = \begin{cases} 1 & \text{if } d = 1 \\ (-1)^k & \text{if } d \text{ が異なる } k \text{ 個の素因数で表される} \\ 0 & \text{if } d \text{ が平方因子を含む} \end{cases}$
I_h	h の約数であるすべての整数の集合
I_h^-	$I_h \setminus \{1\}$
$V(G)$	(ダイ) グラフ G の頂点集合
$E(G)$	グラフ G の辺集合
$A(G)$	ダイグラフ G の弧集合
$\deg v$	v の次数
$\text{id } v$	v の入次数
$\text{od } v$	v の出次数
$\Delta(G)$	G の最大次数
$\delta(G)$	G の最小次数
$\langle U \rangle, U \subset V(G)$	G の U による誘導部分ダイグラフ
$\langle X \rangle, X \subset A(G)$	G の A による弧誘導部分ダイグラフ
$d(u, v)$	u から v への距離
$G_1 \cong G_2$	G_1 と G_2 は同型
$L(G)$	G のラインダイグラフ
$G_1 \otimes G_2$	G_1 と G_2 のクロネッカー積
$G_1 \oplus G_2$	G_1 と G_2 の和
$G_1 \mid G_2$	G_2 は G_1 による同型因子分解を持つ
K_n^*	オーダ n の完全対称ダイグラフ
K_n^+	オーダ n のループ付き完全対称ダイグラフ
$CT(d, h)$	高さ h の完全有向 d 分木
$CT^-(d, h)$	$CT(d, h)$ から $CT(d, h-1)$ の部分木を一つ削除して得られる有向木

$B(d, n)$	直径 n の d 進 de Bruijn ダイグラフ
$K(d, n)$	直径 n の d 進 Kautz ダイグラフ
$G_B(d, n)$	頂点数 n の d 進一般化 de Bruijn ダイグラフ
$G_K(d, n)$	頂点数 n の d 進一般化 Kautz ダイグラフ
$\text{load}_\phi(x)$	ϕ のもとでの頂点 x の負荷
$I_v(m_i, m_j)$	頂点 v における情報 m_i と m_j のインターバル

第3章 de Bruijn 族のグラフ

本研究では de Bruijn ダイグラフおよびその拡張クラスの総称として de Bruijn 族という用語を用いる．本章では代表的な de Bruijn 族のグラフとその性質について紹介する．

3.1 de Bruijn ダイグラフ, Kautz ダイグラフ

de Bruijn 族グラフは de Bruijn 列と呼ばれる巡回列の構成問題を起源としている．de Bruijn 列 $B(d, n)$ は要素数が d であるような与えられたアルファベット A から得られる長さが d^n の巡回列であり, A から構成される長さ n の任意の列がその部分列としてただ一度だけ現れるというものである．このような de Bruijn 列 $B(d, n)$ を構成することは d 進 n 桁の de Bruijn ダイグラフのハミルトニアンサイクルを見つけることに対応する．以降, 本論文では $B(d, n)$ を d 進 n 桁の de Bruijn ダイグラフを表すものとする．de Bruijn ダイグラフおよび Kautz ダイグラフは Degree/Diameter Problem と呼ばれる問題と深く関連している．Degree/Diameter Problem とは (Δ, D) -グラフ問題とも呼ばれる最大次数 Δ と直径 D を持つ最大オーダのグラフを見つけ出す問題である．この値の上界は Moore bound と呼ばれ次の値に等しいことが知られている．

$$1 + \Delta + \Delta(\Delta - 1) + \Delta(\Delta - 1)^2 + \cdots + \Delta(\Delta - 1)^{D-1}.$$

これに近い値を持つ代表的なグラフとして D 次元ハイパーキューブが挙げられるが, de Bruijn ダイグラフおよび Kautz ダイグラフは同じ条件でより多くの頂点を含んでいる．

de Bruijn ダイグラフは文字列, 合同式およびラインダイグラフの三つの定義により表現可能であることが知られている [5].

- 文字列による定義

de Bruijn ダイグラフ $B(d, n)$ は d 種の文字からなる長さ n の文字列を頂点として持つ．頂点 u の後ろから $n - 1$ 文字が頂点 v の前から $n - 1$ 文字と等しいとき, すなわち, $u = u_0, u_1, \dots, u_{n-1}$ と表したとき $v = u_1, \dots, u_{n-1}, \alpha$ と表せるならば, u から v への弧が存在する．ここで, α は d 種の文字のいずれかである．

- 合同式による定義

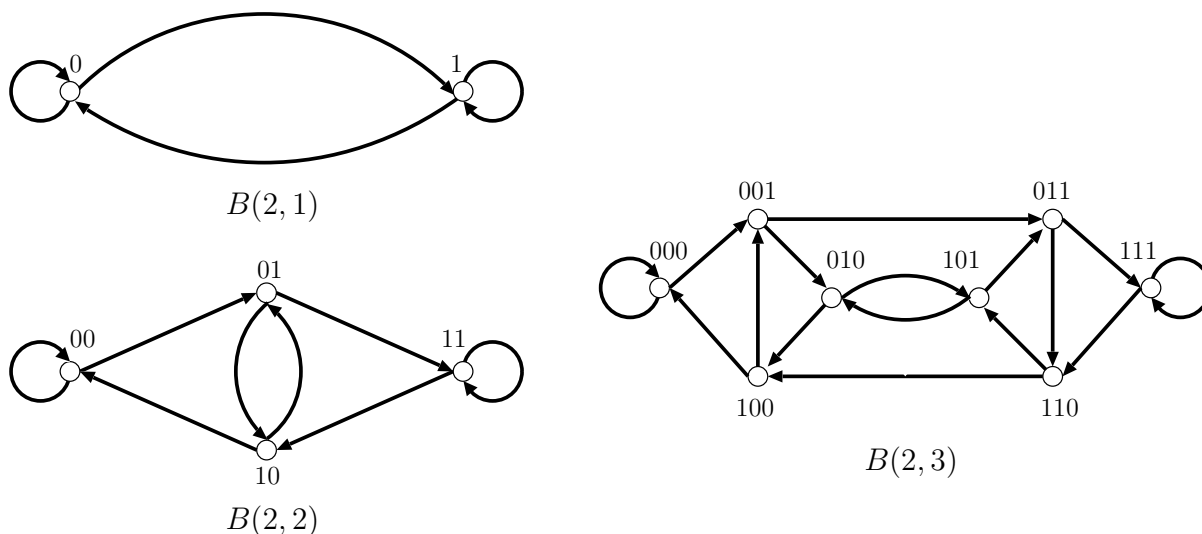


図 3.1: de Bruijn ダイグラフ $B(2, 1), B(2, 2), B(2, 3)$

de Bruijn ダイグラフ $B(d, n)$ に対し, 各頂点を d^n を法とする整数でラベル付けすることで合同式を用いて定義することができる. 頂点 x から頂点 y への弧が存在するための必要十分条件は $y \equiv dx + r \pmod{d^n}$ となるような整数 r ($0 \leq r \leq d - 1$) が存在することである. すなわち,

$$V(B(d, n)) = \mathbb{Z}_{d^n},$$

$$A(B(d, n)) = \{(x, y) \mid y \equiv dx + i \pmod{d^n} \ (0 \leq i < d)\},$$

と定義される.

- ラインダイグラフ演算による定義

de Bruijn ダイグラフの性質として,

$$B(d, n) = L(B(d, n - 1)), \quad n > 1,$$

となることが挙げられる. また, K_d^+ は $B(d, 1)$ と同型である. このことから, 完全ダイグラフ K_d^+ に対してラインダイグラフ演算を $n - 1$ 回行なうことで得られるダイグラフ $L^{n-1}(K_d^+)$ は $B(d, n)$ と同型であることが知られている.

定義より $B(d, n)$ は d 正則であり, 直径が n であることが得られる.

図 3.1 に de Bruijn ダイグラフの例を示す.

Kautz ダイグラフに対しても de Bruijn ダイグラフと同様に三つの定義で表すことができる [5].

- 文字列による定義

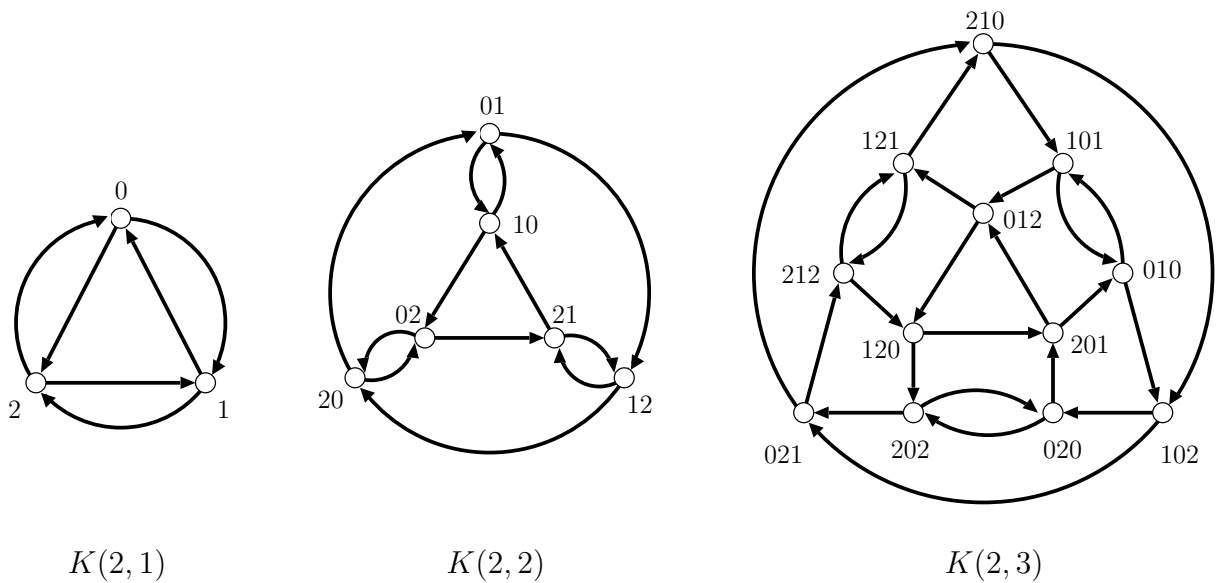


図 3.2: Kautz ダイグラフ $K(2,1), K(2,2), K(2,3)$

Kautz ダイグラフ $K(d, n)$ は $d + 1$ 種の文字からなる長さ n の文字列のうち、どの隣り合う 2 文字も異なるものを頂点として持つ、隣接関係については、頂点 x の後ろから $n - 1$ 文字が頂点 y の前から $n - 1$ 文字と等しいとき、すなわち、 $x = x_0, x_1, \dots, x_{n-1}$ と表したとき $y = x_1, \dots, x_{n-1}, \beta$ と表せるならば、 x から y への弧が存在する。ここで、 β は $d + 1$ 種の文字のうち、 x_{n-1} と異なるいずれかの文字である。

● 合同式による定義

Kautz ダイグラフ $K(d, n)$ に対して、各頂点を $d^n + d^{n-1}$ を法とする整数でラベル付けすることにより合同式で定義することができる。このとき、頂点 x から頂点 y への弧が存在するための必要十分条件は $y \equiv -dx - b \pmod{d^n + d^{n-1}}$ となるような整数 b ($1 \leq b \leq d$) が存在することである。すなわち、

$$V(K(d, n)) = \mathbb{Z}_{d^n + d^{n-1}},$$

$$A(K(d, n)) = \{(x, y) \mid y \equiv -d(x + 1) + i \pmod{d^n + d^{n-1}} \ (0 \leq i < d)\},$$

と定義される。

● ラインダイグラフ演算による定義

Kautz ダイグラフの性質として、

$$K(d, n) = L(K(d, n - 1)), \quad n > 1,$$

となることが挙げられる。また、 K_{d+1}^* は $K(d, 1)$ と同型であることから、 $K(d, n)$ は完全対称ダイグラフ K_{d+1}^* に対してラインダイグラフ演算を $n - 1$ 回行なうことで得られるダイグラフ $L^{n-1}(K_{d+1}^*)$ と同型であることが知られている。

定義より $K(d, n)$ は d 正則であり, 直径が n であることが得られる. 図 3.2 に Kautz ダイグラフの例を示す. Kautz ダイグラフ $K(d, n)$ は de Bruijn ダイグラフ $B(d+1, n)$ の部分グラフであることは文字列の定義より容易に導くことができる.

3.2 一般化 de Bruijn ダイグラフ, 一般化 Kautz ダイグラフ

de Bruijn ダイグラフおよび Kautz ダイグラフは頂点数がそれぞれ $d^n, d^n + d^{n+1}$ として冪乗の形でそれぞれ与えられる. そのため, パラメータ d と n の値を変えることで頂点数に大きな差が生じる. 一般化 de Bruijn ダイグラフおよび一般化 Kautz ダイグラフはこの隔たりを埋めるために頂点数による一般化を行ったダイグラフである.

一般化 de Bruijn ダイグラフは, Reddy, Pradhan and Kuhl [46] と Imase and Itoh [34] で独立に提案されたダイグラフである. このダイグラフは, de Bruijn ダイグラフ $B(d, n)$ の合同式による表現を一般化したものであり, これにより de Bruijn ダイグラフと類似した性質を持つダイグラフを任意の頂点数で定義することが可能となる. 一般化 de Bruijn ダイグラフの定義を次に示す.

定義 3.1 一般化 de Bruijn ダイグラフ $G_B(d, n)$ とは, 頂点集合として 0 から $n-1$ までの整数集合を持ち, 頂点 x が y へ隣接するための必要十分条件は, 合同式

$$y \equiv dx + r \pmod{n}, \quad r = 0, 1, \dots, d-1,$$

を満たす整数 r が存在することであるようなダイグラフである. すなわち,

$$\begin{aligned} V(K(d, n)) &= \mathbb{Z}_n, \\ A(B(d, n)) &= \{(x, y) \mid y \equiv dx + i \pmod{n} \ (0 \leq i < d)\}, \end{aligned}$$

と表せる.

[46] で $G_B(d, n)$ は d 正則であり, $\text{diam}(G_B(d, n)) = \lceil \log_d n \rceil$ であることが示されている. また, ハミルトニアンサイクルが存在する条件についても [18] で与えられており, $d=2$ かつ n が奇数である場合を除いたすべての $G_B(d, n)$ 上にハミルトニアンサイクルが存在する. 図 3.3 に一般化 de Bruijn ダイグラフの例を示す.

一般化 Kautz ダイグラフは Kautz ダイグラフの合同式による表現をしたものであり, [35] で提案された. 定義を次に示す.

定義 3.2 一般化 Kautz ダイグラフ $G_K(d, n)$ とは, 頂点集合として 0 から $n-1$ までの整数集合を持ち, 頂点 x が y へ隣接するための必要十分条件は, 合同式

$$y \equiv -(d+1)x + r \pmod{n}, \quad r = 0, 1, \dots, d-1,$$

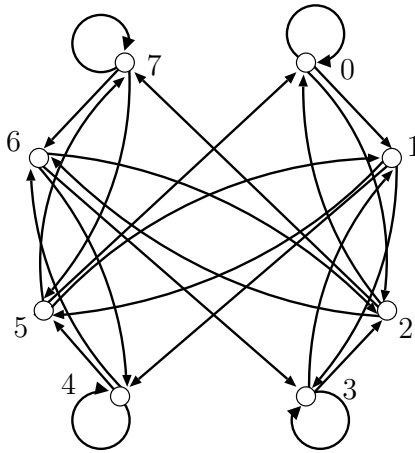


図 3.3: 一般化 de Bruijn ダイグラフ $G_B(3, 8)$

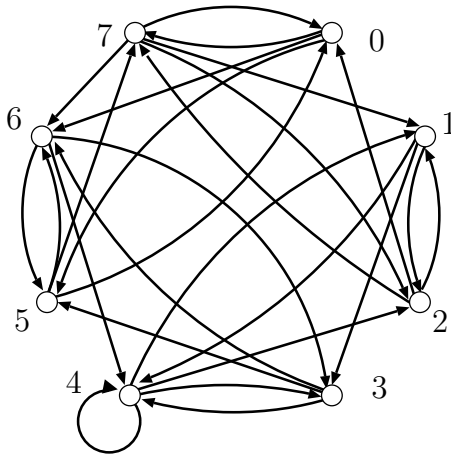


図 3.4: 一般化 Kautz ダイグラフ $G_K(3, 8)$

を満たす整数 r が存在することであるようなダイグラフである。すなわち、

$$V(K(d, n)) = \mathbb{Z}_n,$$

$$A(B(d, n)) = \{(x, y) \mid y \equiv -(d+1)x + i \pmod{n} \ (0 \leq i < d)\},$$

と表せる。

[35] で $G_K(d, n)$ は d 正則であり、 $(G_K(d, n))$ の直径は $\lceil \log_d n \rceil$ か $\lceil \log_d n \rceil - 1$ のいずれかであることが示されている。 $G_K(d, n)$ に対して、 $d = 2$ かつ n が 3^k , $k \geq 1$ と表せない奇数の場合を除いて必ずハミルトニアンサイクルが存在することが知られている [18]。 図 3.4 に一般化 Kautz ダイグラフの例を示す。

第4章 cycle-rooted tree を用いた de Bruijn 族グラフの分解

本章では cycle-rooted tree による de Bruijn 族グラフの分解について述べる．グラフの分解についての研究として，様々なグラフクラスに対し分解に関する結果が得られている．特に，相互結合網のモデルとなるグラフに対して木やサイクルといった情報伝達において自然な構造を持つグラフによる分解を考えることは，情報交換の高速化へとつながるため広く研究されている [6, 12, 30, 39, 51] ．

de Bruijn 族グラフに対しては，de Bruijn ダイグラフおよび Kautz ダイグラフの1因子分解が Bermond と Hell により研究されている [4] ．また，de Bruijn ダイグラフに関しては [3] で木による同型因子分解が，[47] ではハミルトン有向サイクルによる分解が研究されている ．

4.1 de Bruijn ダイグラフの cycle-rooted tree による分解

cycle-rooted tree とは [27, 30] で定義された，有向サイクルを根とする木の形を持つダイグラフである．そのため cycle-rooted tree は情報伝達に適した構造を持っており，それらによる相互結合網の分解を考えることは自然である．de Bruijn 族グラフの cycle-rooted tree による同型因子分解の最初の結果として，Bermond と Fraigniaud により以下が得られている ．

定理 4.1 (Bermond and Fraigniaud [3], Proposition 5.1) de Bruijn ダイグラフ $B(d, n)$ は頂点 (x, x, \dots, x) , $0 \leq x \leq d - 1$ が持つループを root-cycle とするような d 個の完全 d 進 loop-rooted tree により同型因子分解される ．

一方で，[43] では $B(d, n)$ の長さが2以下のすべての有向サイクルに対して高さ $n - 1$ の完全 d 進 loop-rooted tree を構成すると，その各弧集合が $B(d, n)$ の分解であることが示されている．本節ではこれを長さが直径以下の有向サイクルに対して一般化した結果を与える ．

de Bruijn ダイグラフが含む有向サイクルの個数に対して，より一般的な結果として以下が得られている ．

定理 4.2 (Hasunuma and Shibata [29], Theorem 4.5) $m = pd^h$ とおき， $g_l = \gcd(d^l - 1, m)$ とおく．ただし $d \nmid p$ とする．このとき， $p < d^3$ かつ $k \leq \lfloor \log_d m \rfloor + 1$ であるか， $p > d^3$

かつ $k \leq h + 3$ であるならば, $G_B(d, m)$ の長さ k の有向サイクルの総数は

$$\frac{1}{k} \sum_{l|k} \mu\left(\frac{k}{l}\right) g_l \left\lceil \frac{d^l}{g_l} \right\rceil, \quad (4.1)$$

で与えられる.

$B(d, n) \cong G_B(d, d^n)$ であるから, $m = d^n, p = 1, h = n$ とおくことで上の定理が適用できる. このとき, $d > 1$ ならば $p = 1 < d^3$ であり, $k \leq n + 1$ ならば $k \leq \lfloor \log_d m \rfloor + 1 = n + 1$ であるから条件を満たし, 有向サイクルの個数が求められる. 今, 任意の正整数 l に対して $g_l = \gcd(d^l - 1, d^n) = 1$ であるから, 次の系が得られる.

系 4.3 k を $1 \leq k \leq n + 1$ なる正整数とする. このとき $B(d, n)$ における長さ k の有向サイクルの総数は,

$$\frac{1}{k} \sum_{l|k} \mu\left(\frac{k}{l}\right) d^l, \quad (4.2)$$

で与えられる.

$B(d, n)$ において, 長さが $k \leq n$ の有向サイクルに存在する任意の頂点 $v = (v_0, v_1, \dots, v_{n-1})$ の文字列の性質を調べる. 今, 頂点 v を始点とする長さ k の有向サイクルが存在すると仮定すると, v の文字列は自身の文字列を k 桁左に巡回シフトした文字列と等しくなることがわかる. すなわち, $0 \leq i \leq k - 1$ なる i に対し, $v_i = v_{i+k}$ が得られる. これは v_{2k-1} 以降の文字に対しても同様であるため, v の文字列は v_0, v_1, \dots, v_{k-1} が繰り返し現れる形であるから, v の文字列は,

$$v_i = v_{i+mk}, \quad 1 \leq m \leq \left\lfloor \frac{n-1-i}{k} \right\rfloor, \quad (4.3)$$

という関係を満たす. このような部分文字列 v_0, v_1, \dots, v_{k-1} を反復文字列と定義し, 頂点 v は反復文字列 v_0, v_1, \dots, v_{k-1} からなるということにする. このとき, 反復文字列 v_0, v_1, \dots, v_{k-1} は自身の中に $v_{k'-1} = v_{k-1}$ となるような長さ k' の反復文字列を含むことがないことに注意する. これは, v の文字列を $k' - 1 \mid k - 1$ 桁だけ左に巡回シフトすることで v と同じ文字列となり, 長さ k の有向サイクルを構成することに矛盾するためである.

$B(d, n)$ 上の長さが k であるすべての有向サイクルの集合を C_k と定義する. このとき, C_k 内の異なる二つの有向サイクル C_1, C_2 に対し $V(C_1) \cap V(C_2) = \emptyset$ を示すことで, 長さ k の有向サイクルに含まれる頂点の総数が $\sum_{l|k} \mu(k/l) \cdot d^l$ と定まる.

補題 4.4 d, n を $d, n \geq 2$ を満たす正整数とし, k を n 以下の正整数とする. このとき, $B(d, n)$ の C_k 内の異なる二つの有向サイクル C_1, C_2 に対し $V(C_1) \cap V(C_2) = \emptyset$ が成り立つ.

証明 逆に $V(C_1) \cap V(C_2) \neq \emptyset$ であると仮定する．このとき， $V(C_1)$ と $V(C_2)$ のどちらにも含まれるような $V(B(d, n))$ の頂点が少なくとも一つ存在し，その中で C_1 と C_2 でそれぞれ異なる頂点と隣接するものを $v = (v_0, v_1, \dots, v_{n-1})$ として表すことにする． C_1 上で v に隣接される頂点を $u = (v_1, v_2, \dots, v_{n-1}, u_0)$ ， C_2 上で v に隣接される頂点を $w = (v_1, v_2, \dots, v_{n-1}, w_0)$ とおく．このとき， u は長さ k の有向パス上に存在するため $u_0 = v_{n-k}$ が成り立つ．同様に， w は長さ k の有向パス上に存在するため $w_0 = v_{n-k}$ が成り立つ．したがって， $u_0 = w_0$ となり $u = w$ が従うため，仮定に矛盾する． \square

今，頂点 v が長さ $k \leq n$ の有向サイクルに含まれると仮定すると， v は長さ k の反復文字列から成る．このとき v は $k' \mid k$ を満たすような長さ k' の反復文字列から成ることがないため，補題 4.4 は以下のように拡張される．ここで，正整数 k, h に対して， $I_h = \{k \mid k \text{ は } h \text{ の約数}\}$ と定義する．

補題 4.5 h を n 以下の正整数とし， $k_1, k_2 \in I_h$ であるとする．このとき， $B(d, n)$ の C_{k_1} に属する任意の有向サイクル C_1 と C_{k_2} に属する任意の有向サイクル C_2 に対し $V(C_1) \cap V(C_2) = \emptyset$ が成り立つ．

補題 4.4 および補題 4.5 では $B(d, n)$ 上の二つの有向サイクルが持つ頂点の性質を示している．これらの有向サイクルを root-cycle とするような cycle-rooted tree が $B(d, n)$ 内に存在することを次で述べる．

補題 4.6 k を n 以下の正整数とし， C を $B(d, n)$ 上の長さ k の有向サイクルとする．このとき， $(u, v) \in A(C)$ なる頂点 u に対し， u を根とし， v を除く $d-1$ 個の隣接頂点を子として持つ $CT^-(d, n+1-k)$ と同型な木が $B(d, n)$ 上に存在する．

証明 逆を仮定し，長さが $n+1-k$ 以下かつ，弧 (u, v) を含まない二つの異なる $u-w$ 有向パス P_1, P_2 を持つ頂点 w が存在すると仮定する． P_1 の長さを $n-t_1$ ， P_2 の長さを $n-t_2$ とおく．一般性を失うことなく， $n \geq t_1 \geq t_2 \geq k-1$ とおける． $u = (u_0, u_1, \dots, u_{n-1})$ ， $u = (u_1, \dots, u_{n-1}, u_n)$ とおくと， P_1 より w は

$$w = (\underbrace{u_{n-t_1}, u_{n-t_1+1}, \dots, u_{n-1}}_{t_1}, \underbrace{w_0, w_1, \dots, w_{n-t_1-1}}_{n-t_1})$$

と表せる．ただし， $w_i \in \mathbb{Z}_d$ ， $0 \leq i \leq n-t_1-1$ であり， $(u, v) \notin P_1$ より $w_0 \neq u_n$ である．同様に， P_2 より，

$$w = (\underbrace{u_{n-t_2}, u_{n-t_2+1}, \dots, u_{n-1}}_{t_2}, \underbrace{w'_0, w'_1, \dots, w'_{n-t_2-1}}_{n-t_2})$$

と表せる ($w'_i \in \mathbb{Z}_d$ ， $0 \leq i \leq n-t_2-1$ ， $w'_0 \neq u_n$)．今，これらの二つの文字列は等しいので， $u_{n-t_1+i} = u_{n-t_2-i}$ ， $0 \leq i \leq t_2-1$ となる． $t_1 \geq t_2 \geq k-1$ であるから，これらの部分文字

列には v の長さ k の反復文字列 v_0, v_1, \dots, v_{k-1} が少なくとも一回は含まれている．このとき， v 内の文字 v_j , $n - t_2 \leq j \leq n - 1$ に対し， v_j から $t_1 - t_2$ 回ずつ左または右にシフトした桁の文字は v_j の反復として現れる．一方で， v は長さ k の反復文字列を含むので， v_j から k 回ずつ左または右にシフトした桁の文字もまた v_j の反復として現れる．したがって，任意の整数 m_1 と m_2 に対し $v_j = v_{j+m_1k+m_2(t_1-t_2)}$ を満たすため，任意の整数 m に対し， $v_j = v_{j+m \gcd(k, t_1-t_2)}$ が成り立つ． $\gcd(k, t_1 - t_2) \neq k$ と仮定すると， v は長さ $\gcd(k, t_1 - t_2)$ の有向サイクルに含まれることになり， $\gcd(k, t_1 - t_2) \mid k$ であるから補題 4.5 に反し矛盾が生じる．よって， $\gcd(k, t_1 - t_2) = k$ と仮定でき， $k \mid t_1 - t_2$ となるから，ある整数 x に対し $t_1 - t_2 = xk$ とおける．しかしながら， $u_{n-t_1+t_2} = u_{n-xk} = w'_0 \neq u_n$ となり， v が長さ k の反復文字列を含むことに矛盾する．

以上より，長さが $n + 1 - k$ 以下かつ，弧 (u, v) を含まない二つの異なる $u-w$ 有向パスを持つ頂点 w は存在しない． $B(d, n)$ は d 正則であるから， v の $d - 1$ 個の子はそれぞれ点素な高さ $n - k$ の完全 d 分木を持ち， u を根とするような $CT^-(d, n + 1 - k)$ と同型な木が $B(d, n)$ 上に存在する． \square

長さ $k \leq n$ の有向サイクルに含まれる異なる二つの頂点 u, v に対しても，補題 4.6 と同様の方法を用いることでそれぞれが互いに点素な $CT^-(d, n + 1 - k)$ を collateral tree として持つことがいえる．したがって，補題 4.6 から次が得られる．

補題 4.7 k を n 以下の正整数とし， C を $B(d, n)$ 上の長さ k の有向サイクルとする．このとき， C を root-cycle として持つ高さ $n + 1 - k$ の完全 d 進 cycle-rooted tree が $B(d, n)$ 上に存在する．

補題 4.6 および補題 4.7 より，長さ n 以下の任意の有向サイクルを root-cycle とするような完全 d 進 cycle-rooted tree が $B(d, n)$ 上に存在する．一方で，これらの完全 d 進 cycle-rooted tree で $B(d, n)$ が分解できるためにはそれぞれの cycle-rooted tree が辺素の必要がある．まず，異なる二つの有向サイクル上の頂点の距離に関する結果を次で与える．

補題 4.8 h を n 以下の正整数とし， $k_1, k_2 \in I_h$ とする．また， C_1, C_2 をそれぞれ $C_1 \in \mathcal{C}_{k_1}$, $C_2 \in \mathcal{C}_{k_2}$ であるような異なる二つの有向サイクルとする．このとき， $u \in V(C_1)$ である任意の頂点 u と $v \in V(C_2)$ である任意の頂点 v に対し， $d(u, v) \geq n + 1 - h$ かつ $d(v, u) \geq n + 1 - h$ である．

証明 逆を仮定し，今，頂点 u, v に対し $d(u, v) \leq n - h$ であると仮定する． $d(u, v) = l$ とおき， $u = (u_0, u_1, \dots, u_{n-1})$, $v = (v_0, v_1, \dots, v_{n-1})$ とおく．このとき， v の文字列は u の文字列を l 桁左に巡回シフトした文字列であるので， $v_i = u_{l+i}$ ($0 \leq i \leq n - l - 1$) が成り立つ．ここで， u の後ろから $n - l$ 桁の部分文字列 u_l, \dots, u_{n-1} を u' , v の前から $n - l$ 桁の部分文字列 v_0, \dots, v_{n-l-1} を v' とおく．以下の場合を考える．

1. $k_1 = k_2$ の場合

仮定より $l = d(u, v) \leq n - h$ であり, この式を変形すると $h \leq n - l$ を得る. ここで, k_1 は h の約数であるから $k_1 \leq h \leq n - l$ となり, 部分文字列 u' には u の反復文字列が, v' には v の反復文字列がそれぞれ含まれることがいえる. 今, u' 内で u_0 の反復として現れる最初のインデックスを p とおくと, $u_p = u_0, l \leq p \leq l + k_1$ となる. このとき $v_i = u_{l+i}$ であるから, $v_{p-l} = u_p$ となる. さらに, $0 \leq p - l \leq k_1$ であるから v_{p-l} は v の反復文字列の $p - l$ 桁目の値であることがわかる. $v_{p-l} = u_p = u_0$ より v の反復文字列は u の反復文字列を $p - l$ 桁だけ左に巡回シフトしたものと等しくなり, これは u と v を含む長さ k_1 の有向サイクルが存在することを意味している. しかしながら, これは補題 4.4 の長さ k_1 の各有向サイクルの頂点集合は互いに素であることに矛盾するため, $d(u, v) \leq n - h$ となる頂点 u, v は存在しない.

2. $k_1 \neq k_2$ の場合

さらに二つの場合に分けて考える.

(a) $k_1 \mid k_2$ または $k_2 \mid k_1$ の場合

一般性を失うことなく $k_1 \mid k_2$ とし, $k_2 = x \cdot k_1$ とする. u の反復文字列の長さは k_1 なので, (4.3) 式より $u_i = u_{i+m k_1}$ である. また $v' = u'$ より, $v_i = u_{l+i} = u_{l+i+m k_1} = v_{i+m k_1}$ である. これは v の反復文字列の長さが k_1 であることを示し, $k_1 \neq k_2$ に矛盾する.

(b) $k_1 \nmid k_2$ かつ $k_2 \nmid k_1$ の場合

一般性を失うことなく $k_1 > k_2$ とする. また, $k_1 = \alpha \gcd(k_1, k_2), k_2 = \beta \gcd(k_1, k_2)$ とおく. $k_1 \mid h$ かつ $k_2 \mid h$ より $\text{lcm}(k_1, k_2) = \alpha\beta \gcd(k_1, k_2)$ であり, $\text{lcm}(k_1, k_2) \leq h \leq n - l$ であるので, u' と v' において各反復文字列はそれぞれ β 回以上と α 回以上現れる. v' 内の任意の文字を $v_j, 0 \leq j \leq l_2 - 1$ とおく. このとき, v' 内で v_j から k_2 回ずつ左または右にシフトした桁にある文字は v_j の反復として現れる. 今, $v' = u'$ であるから, v' 内で v_j から k_1 回ずつ左または右にシフトした桁にある文字もまた v_j の反復として現れる. したがって, 任意の整数 m_1 と m_2 に対し $v_j = v_{j+m_1 k_1+m_2 k_2}$ を満たす. $k_1 = \alpha \gcd(k_1, k_2), k_2 = \beta \gcd(k_1, k_2)$ より, $m_1 k_1 + m_2 k_2 = (m_1 \alpha + m_2 \beta) \gcd(k_1, k_2)$ と表せるため, $v_{j+(m_1 \alpha+m_2 \beta) \gcd(k_1, k_2)}$ で表される桁は v_j の反復として v 上に現れる. ここで, \gcd の定義より α と β は互いに素であるから, $m_1 \alpha + m_2 \beta$ が任意の整数となるような整数 m_1 と m_2 の値が存在する. したがって, $v_j = v_{j+m \gcd(k_1, k_2)}$ となり, v は長さ $\gcd(k_1, k_2)$ の反復文字列を含むことになるが, これは補題 4.5 に矛盾する.

以上より, $d(u, v) \leq n - h$ を満たすような頂点 u, v は存在せず, 命題は真である. □

補題 4.8 より二つの異なる有向サイクルの頂点間の距離が $n+1-h$ であることが示せた。次に、これらの有向サイクルを root-cycle とするような完全 d 進 cycle-rooted tree に含まれる頂点間の距離についての関係を示す。

補題 4.9 h を n 以下の正整数とし、 $k_1, k_2 \in I_h$ とする。また、 C_1, C_2 をそれぞれ $C_1 \in \mathcal{C}_{k_1}$, $C_2 \in \mathcal{C}_{k_2}$ であるような異なる二つの有向サイクルとし、 CRT_1, CRT_2 をそれぞれ C_1, C_2 を root-cycle とする高さ $n-h$ の完全 d 進 cycle-rooted tree とする。このとき、二つの異なる cycle-rooted tree CRT_1, CRT_2 に対し $V(CRT_1) \cap V(CRT_2) = \emptyset$ が成り立つ。

証明 逆を仮定し、 CRT_1 のある cycle-vertex u からの高さが $n-h$ 以下かつ、 CRT_2 のある cycle-vertex v からの高さが $n-h$ 以下であるような頂点 w が存在すると仮定する。 $d(u, w) = t_1$, $d(v, w) = t_2$ とおく。一般性を失うことなく、 $0 \leq t_1 \leq t_2 \leq n-h$ とおける。 $u = (u_0, u_1, \dots, u_{n-1})$, $v = (v_0, v_1, \dots, v_{n-1})$, $w = (w_0, w_1, \dots, w_{n-1})$ とおくと、 $d(u, w) = t_1$ より

$$w_i = u_{t_1+i}, \quad 0 \leq i \leq n-t_1-1, \quad (4.4)$$

と表せる。同様に、 $d(v, w) = t_2$ より、

$$w_j = v_{t_2+j}, \quad 0 \leq j \leq n-t_2-1, \quad (4.5)$$

と表せる。4.4 式と 4.6 式より、

$$u_{t_1+l} = v_{t_2+l}, \quad 0 \leq l \leq n-t_2-1, \quad (4.6)$$

が得られる。今、 $0 \leq t_1 \leq t_2 \leq n-h$ であるから、これら $n-t_2$ 桁の部分文字列には u および v の反復文字列が少なくとも一回は含まれている。このとき、 u の $n-t_2$ 桁の部分文字列 $u_{t_1}, u_{t_1+1}, \dots, u_{n-1-(t_2-t_1)}$ を u' 、 v の $n-t_2$ 桁の部分文字列 $v_{t_2}, v_{t_2+1}, \dots, v_{n-1}$ を v' とおくことで補題 4.8 と同様の場合分けを用いることができ、同様の結果が得られる。□

$B(d, n)$ の完全 d 進 cycle-rooted tree による分解について、高さ k の完全 d 進 cycle-rooted tree の頂点数は root-cycle の長さ l に関連した値 ld^{k-1} によって与えられる。一方で、 $|V(B(d, n))| = d^n$ であるから、これを高さ k の完全 d 進 cycle-rooted tree により分解しようとするとき、各 root-cycle に含まれる頂点の総和が d の冪として表される必要がある。 $B(d, n)$ 上での長さが n 以下のすべての有向サイクルに含まれる頂点の総数を次で与える。

定理 4.10 h を n 以下の正整数とする。このとき、長さ $k \in I_h$ である $B(d, n)$ 上のすべての有向サイクルに含まれる頂点の総数は

$$\sum_{k \in I_h} \sum_{l|k} \mu\left(\frac{k}{l}\right) \cdot d^l = d^h, \quad (4.7)$$

で表される。

証明 補題 4.4 より，長さ k の二つの有向サイクルは点素であるから，長さ k の有向サイクルに含まれる頂点の総数は有向サイクルの数に k を掛けることで表せる．系 4.3 より，これは $\sum_{l|k} \mu(k/l) \cdot d^l$ で与えられる．同様に補題 4.4 より長さ n 以下である二つの有向サイクルは点素であるから，それらに含まれる頂点の総数は各長さにおける頂点の総数の単純な総和として表せる．すなわち，

$$\sum_{k \in I_h} \sum_{l|k} \mu\left(\frac{k}{l}\right) \cdot d^l,$$

と表せ，これは与式の左辺である．

次に，与えられた等式が正しいことを示す．左辺において k を固定したときに内側の和で l が取り得る値は k のすべての約数である．一方で， l を $l = l'$ として固定した場合， l' を約数として含むすべての k に対してのみ $d^{l'}$ を含む項が存在することは明らかである．したがって，左辺の和の順序を入れ換えると，

$$\begin{aligned} (\text{左辺}) &= \sum_{l|h} \sum_{\substack{k \in I_h \\ l|k}} \mu\left(\frac{k}{l}\right) d^l \\ &= \sum_{l|h} \left(d^l \sum_{k \in I_{h/l}} \mu(k) \right) \\ &= \sum_{l|h} \left(d^l \sum_{k|(h/l)} \mu(k) \right), \end{aligned}$$

と表せる．ここで， $l \neq h$ のとき $\sum_{k|(h/l)} \mu(k) = 0$ であるから，

$$\begin{aligned} &= d^h \sum_{k|1} \mu(k) \\ &= d^h \mu(1) = d^h, \end{aligned}$$

となり，(左辺)=(右辺) が成り立つ． □

系 4.3 より， $B(d, n)$ において root cycle の長さが i である cycle-rooted tree の数は求められ，この値を s_i と表す．また， $B(d, n)$ 上の root-cycle の長さが i である高さ l の完全 d 進 cycle-rooted tree の集合を $\{CRT_1^{i,l}, CRT_2^{i,l}, \dots, CRT_{s_i}^{i,l}\}$ ， $CRT^{i,l} = \bigoplus_{1 \leq k \leq s_i} CRT_k^{i,l}$ と定義する．

$B(d, n)$ の cycle-rooted tree による分解について，次に示す．

定理 4.11 h を n 以下の正整数とする．このとき，

$$B(d, n) = \bigoplus_{i \in I_h} CRT^{i, n+1-h}$$

が成り立つ．

証明 補題 4.5 および定理 4.10 より，長さが h の約数である任意の二つの有向サイクルは点素であり，有向サイクルに含まれる頂点の総数は d^h である．さらに補題 4.8 より，それら二つの有向サイクル上の各頂点間の距離は $n+1-h$ 以上であり，補題 4.7 および補題 4.9 よりこれらの有向サイクルを root-cycle とした深さ $n-h$ の完全 d 進 cycle-rooted tree は点素であり， $n+1-h$ の完全 d 進 cycle-rooted tree は弧素である．完全 d 進 cycle-rooted tree は各 cycle-vertex が $d-1$ 個の子へ隣接し，cycle-vertex でも葉でもない頂点は d 個の子へと隣接している．したがって，長さが h の約数である有向サイクルを root-cycle とした深さ $n-h$ の完全 d 進 cycle-rooted tree に含まれる頂点の総数は，

$$\begin{aligned} d^h \left(1 + (d-1) \sum_{i=0}^{n-h} d^i \right) &= d^h (1 + d^{n-h} - 1) \\ &= d^n, \end{aligned}$$

であり，これは $B(d, n)$ の頂点数と等しい．今，深さ $n+1-h$ の完全 d 進 cycle-rooted tree はこれらの頂点すべてから d 個の弧が接続しているので，その総数は d^{n+1} となり， $B(d, n)$ の弧数と等しい．以上より， $B(d, n)$ はこれらの cycle-rooted tree により分解されることが示され，題意を満たす． \square

定理 4.11 に対するダイグラフの coloring を用いた異なる証明を付録として巻末に記載する．

4.2 Kautz ダイグラフの分解

本節では Kautz ダイグラフ $K(d, n)$ の cycle-rooted tree による分解を与える．まず，グラフの coloring を用いた 2-cycle-rooted tree による同型因子分解を示し，root-cycle の長さを $k \leq n$ に拡張した場合の分解を与える．

4.2.1 ダイグラフの coloring

ダイグラフを分解する方法の一つとして，何らかの coloring による集合の分割に基づくものがある．ダイグラフ G の arc-coloring とは G の弧集合から色集合 C への写像 θ であり， G の任意の弧 $(u, v), (w, x) \in A(G)$ に対して $u = w$ または $v = x$ ならば $\theta(u, v) \neq \theta(w, x)$ として定義される．すなわち，各頂点に接続される弧にはそれぞれ異なる色が割り当てられ，同様に各頂点から接続する弧もそれぞれ異なる色が割り当てられるような coloring である．

Kawai らは [39] で arc-coloring の概念を拡張した新たな arc-coloring の手法を定義している．

定義 4.12 (Kawai, Fujikake and Shibata [39]) ダイグラフ G の弧集合 $A(G)$ から色集合 C への写像を θ とおく． $(u, v), (w, v) \in A(G)$ であるようなすべての頂点 $u, v, w \in V(G)$

の組に対し, $\theta(u, v) \neq \theta(w, v)$ を満たすとき, θ はダイグラフ G の in-coloring であるという.

定義 2.1 と定義 4.12 より次の定理が導かれる.

定理 4.13 (Kawai, Hujikake and Shibata [39], Proposition 8) G を d 色で in-coloring された d 入正則ダイグラフとする. 各色で色付けされた弧は G の因子として d 個の cycle-rooted tree を誘導する.

d 入正則ダイグラフ G 上での in-coloring は $V(G)$ から $V(G)$ への関数の集合と同一とみなすことができる. すなわち, G の任意の 2 頂点 u, v に対し, $(u, v) \in A(G)$ であるための必要十分条件が $u = \lambda_i(v)$ であり, かつ $\lambda_i(v) \neq \lambda_j(v)$ ($i \neq j$) を満たす関数の集合 $\Lambda = \{\lambda_0, \lambda_1, \dots, \lambda_{d-1}\}$ が存在することである.

さらに, ダイグラフ G 上の分解を保存するようなラインダイグラフ $L(G)$ の in-coloring が定義されている.

定義 4.14 (Kawai, Hujikake and Shibata [39]) G を $\Lambda = \{\lambda_0, \lambda_1, \dots, \lambda_{d-1}\}$ によって in-coloring された d 入正則ダイグラフとする. このとき,

$$\lambda_i^* : (u, v) \mapsto (\lambda_i(u), u),$$

を満たすような $L(G)$ の in-coloring $\Lambda^* = \{\lambda_0^*, \lambda_1^*, \dots, \lambda_{d-1}^*\}$ を $L(G)$ の grown coloring と呼ぶ.

grown coloring を繰り返し適用することによって, 反復ラインダイグラフに対する grown coloring が得られる.

定義 4.15 (Kawai, Hujikake and Shibata [39]) G を $\Lambda = \{\lambda_0, \lambda_1, \dots, \lambda_{d-1}\}$ によって in-coloring された d 入正則ダイグラフとする. $k \geq 1$ に対して, $\Lambda^* = \{\lambda_0^*, \lambda_1^*, \dots, \lambda_{d-1}^*\}$ を

$$\lambda_i^* : (x_0, x_1, \dots, x_k) \mapsto (\lambda_i(x_0), x_0, \dots, x_{k-1}),$$

となるような $V(L^k(G))$ から $V(L^k(G))$ への関数の集合とする. このとき, Λ^* は $L^k(G)$ の in-coloring であり, $L^k(G)$ の k -iterated grown coloring と呼ばれる.

4.2.2 2-cycle-rooted tree を用いた同型因子分解

Kautz ダイグラフに対し, k -iterated grown coloring を適用することで, 次の結果が得られる.

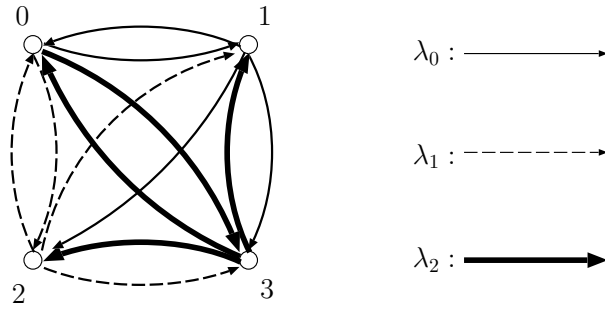


図 4.1: K_4^* の grown coloring の例

定理 4.16 (Kawai, Hujikake and Shibata [39], Corollary 21) Kautz ダイグラフ $K(d, n)$ は root-cycle の長さが 2 であり, 一方の cycle-vertex が持つ collateral tree が $CT^-(d, n)$ と同型であり他方が持つ collateral tree が $CT^-(d, n - 1)$ と同型であるような完全 d 進 2-cycle-rooted tree によって同型因子分解される .

この定理は K_{d+1}^* の in-coloring として $0 \leq i \leq d-1$ に対し, $x \neq i+1$ のとき $\lambda_i(x) = i+1$, $x = i+1$ のとき $\lambda_i(x) = 0$ となるような関数の集合 $\Lambda = \{\lambda_0, \lambda_1, \dots, \lambda_{d-1}\}$ を与え, この in-coloring に対し $L^{n-1}(K_{d+1}^*)$ の $(n-1)$ -iterated grown coloring を行なうことで $K(d, n)$ を cycle-rooted tree で同型因子分解している . 図 4.1 に, grown coloring を用いた K_4^* の同型因子分解を示す .

定理 4.16 をもとに前述の K_{d+1}^* の in-coloring から得られる k -iterated grown coloring を用いて $K(d, n)$ の同型因子分解を行なうと, cycle-vertex が 0 と i ($1 \leq i \leq d$) の交互列である二つの頂点からなる d 個の cycle-rooted tree が因子として得られる . これらの因子を上述の i の値に応じて F_i とし, 各 F_i が持つ二つの cycle-vertex のうち, 末尾が i である頂点を dv_i , 末尾が 0 である頂点を sv_i と定める . さらに, dv_i を根とする collateral tree を $Co-dT_i$, sv_i を根とする collateral tree を $Co-sT_i$ と呼ぶことにする . 各 F_i に対し, $Co-dT_i$ は $CT^-(d, n)$ と同型であり, $Co-sT_i$ は $CT^-(d, n - 1)$ と同型である . したがって, この方法で得られる cycle-rooted tree の葉以外の出次数はすべて d である . 図 4.2 に $K(d, n)$ を同型因子分解した場合の各因子の構造を示す .

また, 図 4.3 に $K(3, 3)$ の cycle-rooted tree による同型因子分解の例を示す . これは K_4^* に 2-iterated grown coloring を行なうことによって得られる同型因子分解である ,

4.2.3 cycle-rooted tree による分解の一般化

Kautz ダイグラフ $K(d, n)$ に対しても de Bruijn ダイグラフと同様に任意の長さの root-cycle を持つ完全 d 進 cycle-rooted tree による分解を与えることができる . 本小節では 4.1 と同様の方法を用いた $K(d, n)$ の分解を与える . ただし, $K(d, n)$ は $B(d, n)$ と違いループ頂点を持たないことに注意する .

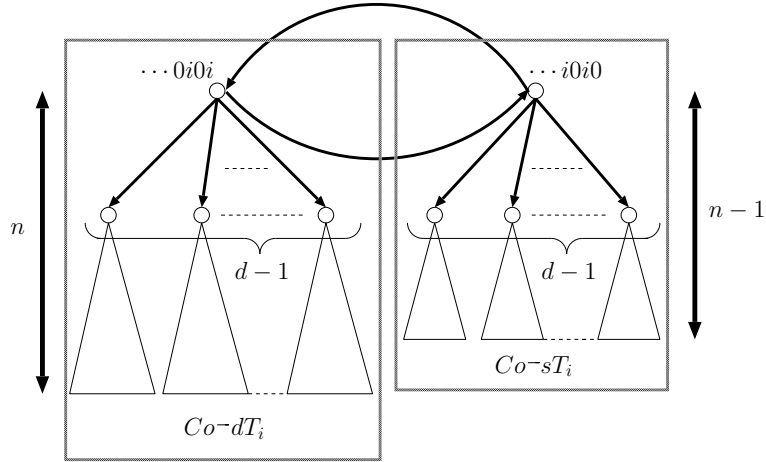


図 4.2: Kautz ダイグラフ $K(d, n)$ の同型因子分解における各因子の構造

一般化 Kautz ダイグラフ上での有向サイクルの個数は次で与えられている .

定理 4.17 (Hasunuma, Kikuchi, Mori and Shibata [28], Proposition 5.1) $d \nmid p$ であるような p に対し , $m = pd^h$ とし , $g_l = \gcd(d^l - (-1)^l, m)$ とおく . このとき , 次の条件

- $p \leq \sqrt{\frac{d^7}{d+1}}$,
- $\sqrt{\frac{d^7}{d+1}} < p < d^5(d+1)$ かつ $k \leq \log_d \left(\frac{n}{\sqrt[3]{p^2(d+1)}} \right) + \frac{10}{3}$,
- $d^5(d+1) < p$ かつ $k \leq \log_d \left(\frac{n}{d+1} \right)$,

のいずれかを満たすとき , $G_K(d, m)$ の長さ k の有向サイクルの総数は

$$\frac{1}{k} \sum_{l|k} \mu \left(\frac{k}{l} \right) g_l (d^l + (-1)^l) (g_l - 1), \quad (4.8)$$

によって与えられる .

$K(d, n) \cong G_K(d, d^n + d^{n-1})$ であるから , $m = d^n + d^{n-1}, p = d + 1, h = n - 1$ とおくことで上の定理が適用できる . このとき , $d > 1$ ならば $p = d + 1 < \sqrt{d^7/(d+1)}$ であるから条件を満たし , 有向サイクルの個数が求められる . 任意の正整数 l に対して $g_l = \gcd(d^l - 1, d^n + d^{n-1}) = d + 1$ を考える . l が偶数 , すなわち $l = 2x, m \geq 1$ のとき

$$g_l = \gcd(d^{2x} - 1, d^n + d^{n-1}) = \gcd \left((d+1) \sum_{i=0}^{2x-1} d^i, (d+1)d^n \right) = d + 1,$$

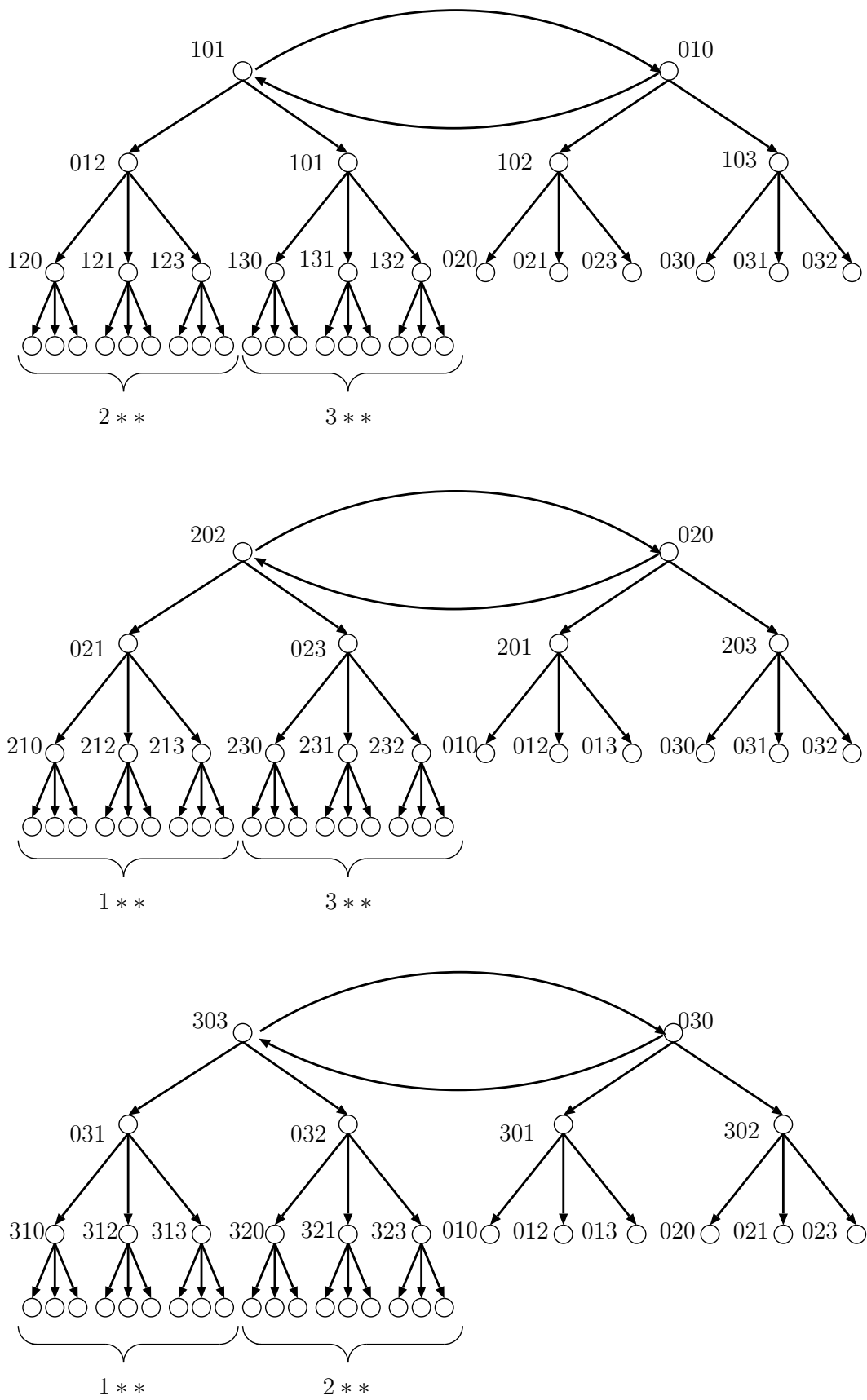


図 4.3: Kautz ダイグラフ $K(3,3)$ の同型因子分解

であり, l が奇数, すなわち $l = 2x + 1$ のとき

$$g_l = \gcd(d^{2x+1} - 1, d^n + d^{n-1}) = \gcd((d+1) \sum_{i=0}^{2x} (-d)^i, (d+1)d^n) = d+1,$$

であるから, $g_l = d+1$ が得られる. この値を用いることで次の系を得る.

系 4.18 k を $1 \leq k \leq n$ なる正整数とする. このとき, $K(d, n)$ における長さ k の有向サイクルの総数は

$$\frac{1}{k} \sum_{l|k} \mu\left(\frac{k}{l}\right) (d^l + (-1)^l d), \quad (4.9)$$

で与えられる.

$K(d, n)$ 上の長さが k であるすべての有向サイクルの集合を C_k と定義する. $K(d, n)$ は $B(d+1, n)$ の部分ダイグラフであるから, 4.1 小節のいくつかの結果を同様に与えることができる. 4.1 小節で定義した I_h に対して, $I_h^- = I_h \setminus \{1\}$ と新たに定義する.

補題 4.19 h を $1 < h \leq n$ を満たす正整数とし, $k_1, k_2 \in (I_h^- \cup I_{h-1}^-)$ とする. このとき, $K(d, n)$ の C_{k_1} に属する任意の有向サイクル C_1 と C_{k_2} に属する任意の有向サイクル C_2 に対し $V(C_1) \cap V(C_2) = \emptyset$ が成り立つ.

証明 $K(d, n)$ は $B(d+1, n)$ の部分ダイグラフであるから, $k_1, k_2 \in I_h^-$ であるか, $k_1, k_2 \in I_{h-1}^-$ である場合には補題 4.5 より $V(C_1) \cap V(C_2) = \emptyset$ を満たす.

今, $k_1 \in I_h^-$ とし, $k_2 \in I_{h-1}^-$ とする. 長さ k_1 の有向サイクルを C_1 と長さ k_2 の有向サイクル C_2 のどちらにも含まれるような $K(d, n)$ の頂点を $v = (v_0, v_1, \dots, v_{n-1})$ と仮定する. このとき, v は長さ k_1 の反復文字列を持ち, これを h/k_1 回繰り返すことで $(v_0, v_1, \dots, v_{h-1}, v_h) = (v_0, v_1, \dots, v_{k_1-1}, v_{k_1})$ となる. 同様に, v は長さ k_2 の反復文字列を持つので, $(v_0, v_1, \dots, v_{h-2}, v_{h-1}, v_h) = (v_0, v_1, \dots, v_{k_2-2}, v_{k_2-1}, v_0)$ となる. しかしながら, $v_{h+1} = v_{k_1+1} = v_0$ であり, $v_h = v_0$ となり, $K(d, n)$ の定義よりこのような頂点 v は存在しない. よって $V(C_1) \cap V(C_2) = \emptyset$ が得られる. \square

$K(d, n)$ は $B(d+1, n)$ の部分ダイグラフであるから, 補題 4.6 および補題 4.7 より長さ k ($1 \leq k \leq h$) の有向サイクルを root-cycle として持つ高さ $n+1-k$ の完全 d 進 cycle-rooted tree が $K(d, n)$ 内に存在する. 以下でこれらの cycle-rooted tree が点素であることを示す.

補題 4.20 h を $1 < h \leq n$ を満たす正整数とし, $k_1, k_2 \in (I_h^- \cup I_{h-1}^-)$ であるとする. また, C_1, C_2 をそれぞれ $C_1 \in C_{k_1}$, $C_2 \in C_{k_2}$ であるような異なる二つの有向サイクルとする. このとき, $u \in V(C_1)$ である任意の頂点 u と $v \in V(C_2)$ である任意の頂点 v に対し, $d(u, v) \geq n+1-h$ かつ $d(v, u) \geq n+1-h$ である.

証明 $K(d, n)$ は $B(d+1, n)$ の部分ダイグラフであるから, $k_1, k_2 \in I_h^-$ であるか, $k_1, k_2 \in I_{h-1}^-$ の約数である場合には補題 4.8 より $V(C_1) \cap V(C_2) = \emptyset$ を満たす.

今, $k_1 \in I_h^-$ とし, $k_2 \in I_{h-1}^-$ とする. また, $u = (u_0, u_1, \dots, u_{n-1}) \in C_1, v = (v_0, v_1, \dots, v_{n-1}) \in C_2$ とおく. $d(u, v) = l$ とおき, $l \leq n-h$ であると仮定すると, $v_i = u_{l-1+i}$ ($0 \leq i \leq n-l-1$) が成り立つ. このとき, $l \leq n-h$ より $h \leq n-l$ であり, v には u の長さ k_1 の反復文字列が少なくとも h/k_1 回含まれることになる. これは補題 4.19 の証明から $v_{h+1} = v_h = v_0$ を導き, $K(d, n)$ の定義に矛盾する. $d(v, u) \leq n-h$ の場合についても同様の矛盾を導く. \square

補題 4.9 および補題 4.20 より次が得られる.

補題 4.21 h を $1 < h \leq n$ を満たす正整数とし, $k_1, k_2 \in (I_h^- \cup I_{h-1}^-)$ であるとする. また, C_1, C_2 をそれぞれ $C_1 \in \mathbf{C}_{k_1}, C_2 \in \mathbf{C}_{k_2}$ であるような異なる二つの有向サイクルとし, CRT_1, CRT_2 をそれぞれ C_1, C_2 を root-cycle とする高さ $n-h$ の完全 d 進 cycle-rooted tree とする. このとき, 二つの異なる cycle-rooted tree CRT_1, CRT_2 に対し $V(CRT_1) \cap V(CRT_2) = \emptyset$ が成り立つ.

$|V(K(d, n))| = d^n + d^{n-1}$ であるから, これを高さ k の完全 d 進 cycle-rooted tree により分解しようとするとき, 各 root-cycle に含まれる頂点の総和が d の冪と $d+1$ の積により表される必要がある. 今, $K(d, n)$ 上での長さが n 以下の有向サイクルに含まれる頂点の総数について, 次を示す.

定理 4.22 h を $h \leq n$ なる正整数とする. このとき, 長さ $k \in (I_h^- \cup I_{h-1}^-)$ である $K(d, n)$ 上のすべての有向サイクルに含まれる頂点の総数は

$$\sum_{k \in I_h^-} \sum_{l|k} \mu\left(\frac{k}{l}\right) \cdot (d^l + (-1)^l d) + \sum_{k \in I_{h-1}^-} \sum_{l|k} \mu\left(\frac{k}{l}\right) \cdot (d^l + (-1)^l d) = d^h + d^{h-1}, \quad (4.10)$$

で表される.

証明 補題 4.19 より長さ $k \in (I_h^- \cup I_{h-1}^-)$ である二つの有向サイクルは点素であるから, それらに含まれる頂点の総数は各長さにおける頂点の総数の単純な総和として表せる. すなわち,

$$\sum_{k \in I_h^-} \sum_{l|k} \mu\left(\frac{k}{l}\right) \cdot (d^l + (-1)^l d) + \sum_{k \in I_{h-1}^-} \sum_{l|k} \mu\left(\frac{k}{l}\right) \cdot (d^l + (-1)^l d),$$

と表せ, これは与式の左辺である.

次に, 与えられた等式が正しいことを示す. 左辺の二つの項それぞれについて, k を固定したときに内側の和で l が取り得る値は k のすべての約数である. 一方で, l を $l = l'$ とし

て固定した場合, l' を約数として含むすべての k に対してのみ $d^{l'}$ を含む項が存在することは明らかである. したがって, 左辺の和の順序を入れ換えると

$$\begin{aligned}
(\text{左辺}) &= \sum_{l|h} \sum_{\substack{k \in I_h \\ l|k}} \mu\left(\frac{k}{l}\right) (d^l + (-1)^l d) + \sum_{l|h-1} \sum_{\substack{k \in I_{h-1} \\ l|k}} \mu\left(\frac{k}{l}\right) (d^l + (-1)^l d) \\
&= \sum_{l|h} \left((d^l + (-1)^l d) \sum_{k \in I_{h/l}} \mu(k) \right) + \sum_{l|h-1} \left((d^l + (-1)^l d) \sum_{k \in I_{h-1/l}} \mu(k) \right) \\
&= \sum_{l|h} \left((d^l + (-1)^l d) \sum_{k|(h/l)} \mu(k) \right) + \sum_{l|h-1} \left((d^l + (-1)^l d) \sum_{k|(h-1/l)} \mu(k) \right),
\end{aligned}$$

と表せる. ここで, $l \neq h$ のとき $\sum_{k|(h/l)} \mu(k) = 0$ であるから,

$$\begin{aligned}
&= (d^h + (-1)^h d) \sum_{k|1} \mu(k) + (d^{h-1} + (-1)^{h-1} d) \sum_{k|1} \mu(k) \\
&= (d^h + (-1)^h d) \mu(1) + (d^{h-1} + (-1)^{h-1} d) \mu(1) = d^h + d^{h-1},
\end{aligned}$$

となり, (左辺)=(右辺) が成り立つ. □

系 4.18 より, $K(d, n)$ において root-cycle の長さが i である cycle-rooted tree の数は求められ, この値を s_i と表す. また, $K(d, n)$ 上の root-cycle の長さが i である高さ l の完全 d 進 cycle-rooted tree の集合を $\{CRT_1^{i,l}, CRT_2^{i,l}, \dots, CRT_{s_i}^{i,l}\}$, $CRT^{i,l} = \bigoplus_{1 \leq k \leq s_i} CRT_k^{i,l}$ と定義する.

$K(d, n)$ の cycle-rooted tree による分解について, 次を示す.

定理 4.23 h を n 以下の正整数とする. このとき,

$$K(d, n) = \bigoplus_{i \in I_h^-} CRT^{i, n+1-h} \bigoplus_{j \in I_{h-1}^-} CRT^{j, n+1-h}$$

が成り立つ.

証明 補題 4.19 および定理 4.22 より, 長さが h の約数である任意の二つの有向サイクルは点素であり, 有向サイクルに含まれる頂点の総数は $d^h + d^{h-1}$ である. さらに補題 4.20 より, それら二つの有向サイクル上の各頂点間の距離は $n+1-h$ 以上である. したがって, これらの有向サイクルを root-cycle とした深さ $n-h$ の完全 d 進 cycle-rooted tree は点素であり, 深さ $n+1-h$ の完全 d 進 cycle-rooted tree は弧素である. 完全 d 進 cycle-rooted tree は各 cycle-vertex が $d-1$ 個の子へ隣接し, cycle-vertex でも葉でもない頂点は d 個の子へと隣接している. したがって, 長さが h の約数である有向サイクルを root-cycle とし

た深さ $n - h$ の完全 d 進 cycle-rooted tree に含まれる頂点の総数は,

$$\begin{aligned} (d^h + d^h - 1) \left(1 + (d - 1) \sum_{i=0}^{n-h} d^i \right) &= (d^h + d^h - 1)(1 + d^{n-h} - 1) \\ &= d^n + d^{n-1}, \end{aligned}$$

であり, これは $K(d, n)$ の頂点数と等しい. 今, 深さ $n + 1 - h$ の完全 d 進 cycle-rooted tree はこれらの頂点すべてから d 個の弧が接続しているので, その総数は $d^{n+1} + d^n$ となり, $K(d, n)$ の弧数と等しい. 以上より, $K(d, n)$ はこれらの cycle-rooted tree により分解されることが示され, 題意を満たす. \square

図 4.4 に $K(2, 4)$ における $h = 4$ の場合の cycle-rooted tree による分解を示す.

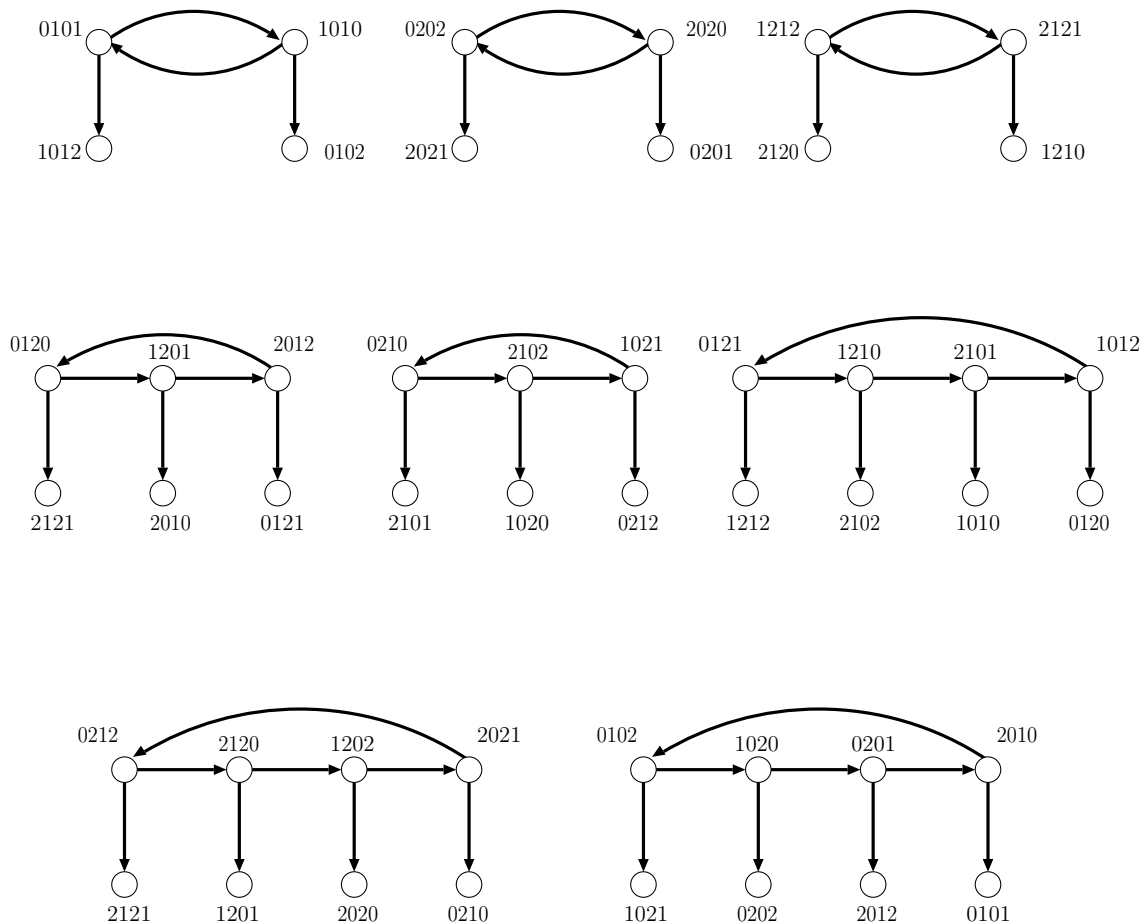


図 4.4: $K(2, 4)$ の分解例

4.3 一般化 de Bruijn および一般化 Kautz ダイグラフの分解

本節では，一般化 de Bruijn ダイグラフおよび一般化 Kautz ダイグラフ上に必ず存在するループについて注目し，ループを root-cycle とする cycle-rooted tree によるいくつかの分解方法について述べる．ここで扱う $G_B(d, n)$, $G_K(d, n)$ は弧の重複を防ぐため $n > d$ の場合のみとする．

$G_B(d, n)$ の頂点 i に対し，弧 $(i, di + k) \pmod{n}$ ($k = 0, 1, \dots, d-1$) をそれぞれ k -value を持つ弧と呼び， $e(i; k)$ と表す．また， $e(i; k)$ から接続されている頂点を $v(i; k)$ と表す． $G_K(d, n)$ に対しても弧 $(i, -d(i+1) + k) \pmod{n}$ について同様の表現を用いる．

一般化 de Bruijn ダイグラフ $G_B(d, n)$ に対し，[52] より次が示されている．

定理 4.24 (Shibata, Shirahata and Osawa [52], Theorem 4) 一般化 de Bruijn ダイグラフ $G_B(d, n)$ が持つ長さ k の閉ウォークの総数は $d^k - 1 + g_k$ によって与えられる．ただし， $g_k = \gcd(d^k - 1, n)$ である．

定理 4.24 より，長さ 1 の閉ウォーク，すなわち，ループの総数は $g = \gcd(d-1, n)$ とおくことで $d-1 + g$ と表せる． $G_B(d, n)$ において， $d-1 = gs_g$, $n = gt_g$ とおくと，各ループ頂点の持つ値は次のように得られる．

定理 4.25 $G_B(d, n)$ に対し，ループ頂点 u は

$$u \equiv yt_g + x \frac{\alpha t_g - 1}{s_g} \pmod{n}, \quad y = 0, 1, \dots, g-1, \quad x = 0, 1, \dots, s_g, \quad (4.11)$$

で表される．ただし， α は $\alpha t_g \equiv 1 \pmod{s_g}$ を満たす最小の整数とする．

証明 定義より， u から隣接している頂点を v とおくと，

$$\begin{aligned} v &\equiv du + r \\ &\equiv d \left(yt_g + x \frac{\alpha t_g - 1}{s_g} \right) + r \pmod{n}, \quad r = 0, 1, \dots, d-1, \end{aligned}$$

と表せる．今，

$$\begin{aligned} v &= d \left(yt_g + x \frac{\alpha t_g - 1}{s_g} \right) + r \\ &= dyt_g + dx \frac{\alpha t_g - 1}{s_g} + r \\ &= (d-1)yt_g + yt_g + (d-1)x \frac{\alpha t_g - 1}{s_g} + x \frac{\alpha t_g - 1}{s_g} + r \\ &= s_g ygt_g + yt_g + \alpha xgt_g - gx + x \frac{\alpha t_g - 1}{s_g} + r \\ &\equiv yt_g + x \frac{\alpha t_g - 1}{s_g} - gx + r \pmod{n} \\ &\equiv u - gx + r, \end{aligned}$$

であるから, $r = gx$ のとき $v = u$ となり u はループを持つ.

次に, (4.11) により得られる各値が n を法として合同でないことを示す. 二つの頂点 $u_1 = y_1 t_g + x_1 \frac{\alpha t_g - 1}{s_g}$, $u_2 = y_2 t_g + x_2 \frac{\alpha t_g - 1}{s_g}$ が, $x_1 \neq x_2$ または $y_1 \neq y_2$ かつ n を法として合同の値を持つと仮定すると,

$$\begin{aligned} y_1 t_g + x_1 \frac{\alpha t_g - 1}{s_g} &\equiv y_2 t_g + x_2 \frac{\alpha t_g - 1}{s_g} \\ (y_1 - y_2) t_g + (x_1 - x_2) \frac{\alpha t_g - 1}{s_g} &\equiv 0 \pmod{n}, \end{aligned} \quad (4.12)$$

が成り立たなければならない. 今, 左辺第一項および n は t_g の倍数であるから, この式が成り立つには左辺第二項も t_g の倍数でなければならない. まず左辺第二項が 0, すなわち $x_1 = x_2$ である場合, $y_1 - y_2$ は g の倍数となる. しかし $0 \leq y_1, y_2 \leq g - 1$ であるから, g の倍数となるのは $y_1 = y_2$ の場合のみであり, これは整数上において同じ値である. 次に, 左辺第二項が 0 でない t_g の倍数の場合を考える. このとき $(x_1 - x_2) \frac{\alpha t_g - 1}{s_g} = m t_g$ と仮定すると, 式変形により $\{(x_1 - x_2)\alpha - s_g m\} t_g = x_2 - x_1$ が得られる. $0 \leq x_1, x_2 \leq s_g$ かつ $x_1 \neq x_2$ であるから, 上の式を満たすには $t_g < |x_1 - x_2|$ を満たす必要がある. しかしながら $t_g > s_g$ であるためそのような x_1, x_2 は存在しない. したがって, $x_1 \neq x_2$ または $y_1 \neq y_2$ を満たす異なる二つの値は n を法として合同ではない. よってこれらの値を持つ頂点 u_1, u_2 は $G_B(d, n)$ 上で異なる頂点であることがいえる. また, このようにして得られるループの数は $g(s_g + 1) = d - 1 + g$ 個であり, $G_B(d, n)$ のループの総数と一致する. \square

一般化 Kautz ダイグラフ $G_K(d, n)$ に対しても同様に頂点の値を得る. まず, $G_K(d, n)$ の閉ループの総数は [28] により得られている.

定理 4.26 (Hasunuma, Kikuchi, Mori and Shibata [28], Theorem 3.2) 一般化 Kautz ダイグラフ $G_K(d, n)$ が持つ長さ k の閉ウォークの総数は $d^k + (-1)^k (h_k - 1)$ によって与えられる. ただし, $h_k = \gcd(d^k - (-1)^k, n)$ である.

定理 4.26 より, $G_K(d, n)$ のループの総数は $h = \gcd(d + 1, n)$ とおくと $d + 1 - h$ で表せる. $G_K(d, n)$ において, $d + 1 = h s_h$, $n = h t_h$ とおくと $G_B(d, n)$ の場合と同様の方法で以下で示される頂点がループを持つことが得られる.

定理 4.27 $G_K(d, n)$ に対し, ループ頂点 u は

$$u \equiv y t_h + x \frac{\alpha t_h - 1}{s_h} + \beta \pmod{n}, \quad y = 0, 1, \dots, g - 1, \quad x = 0, 1, \dots, s_h - 2,$$

で表される. ただし, α は $s_h | \alpha t_h - 1$ を満たす最小の整数であり, β は $t_h | \beta s_h + 1$ を満たす最小の整数である.

$G_K(d, n)$ においては $h = d + 1$ のときループを持たないことに注意する.

4.3.1 loop-rooted tree による因子分解

この小節では $G_B(d, n)$ 及び $G_K(d, n)$ が loop-rooted tree で因子分解可能であるための必要十分条件を与える。

$G_B(d, n)$ および $G_K(d, n)$ はそれぞれ d 正則であり, 各頂点は連続した値を持つ d 個の頂点に隣接しているので, $d < n$ のとき各頂点は異なる d 個の頂点に隣接されている。

$g = 1$ のとき, 定理 4.24 によりループの総数は d 個である。これら d 個のループ頂点を l_0, l_1, \dots, l_{d-1} ($l_0 < l_1 < \dots < l_{d-1}$) とおき, $G_B(d, n)$ 上で 0 へ隣接する頂点を x_0, x_1, \dots, x_{d-1} , ($x_0 < x_1 < \dots < x_{d-1}$) とおく。このとき, 次が成り立つ。

補題 4.28 $x_0 = l_0 = 0, x_{d-1} < l_{d-1} = n - 1$ であり, i ($0 < i < d - 1$) に対し $x_i \leq l_i \leq x_{i+1}$ を満たす。

証明 $x_0 = l_0 = 0, x_{d-1} < l_{d-1} = n - 1$ は明らかである。 $x_i \leq l_i \leq x_{i+1}$ であることを示す。 x_i から x_{i+1} までのどの頂点もループを持たないと仮定する。今, 与えられた範囲内の頂点 v が自身以上の値を持つ頂点へ隣接しているとする, x_i が 0 と隣接しており, 範囲内の頂点は連続であるので v までのいずれかの頂点はループを持ち矛盾が生じる。このことから, 範囲内のどの頂点に対しても自身以上の値を持つ頂点へ隣接することがない。しかしながら x_{i+1} は 0 へ隣接しているため, x_{i+1} と $x_{i+1} - 1$ のどちらかは $n - 1$ と隣接している。各頂点の値は $n - 1$ 以下であるから, 必ずどこかにループを持つ頂点が少なくとも一つ存在する。さらにこれはすべての i に対して成り立つので, 鳩の巣原理よりこのようなループを持つ頂点は x_i と x_{i+1} の間にただ一つ存在する。□

各 l_i に対し, p_i をループとなる弧の k -value として定義し, 各 x_i から 0 へ隣接する弧の k -value を q_i と定義する。これらを元に, 実際に $G_K(d, n)$ の loop-rooted tree による因子分解を与える。

定理 4.29 d, n を $d < n$ を満たす二つの自然数とし, $g = \gcd(d - 1, n)$ とする。このとき, $G_B(d, n)$ が loop-rooted tree により因子分解可能であるための必要十分条件は $g = 1$ であることである。

証明 $g > 1$ と仮定する。このとき, $G_B(d, n)$ のループ数は $d - 1 + g$ であるから $G_B(d, n)$ 上には d 個より多くのループが存在する。 $G_B(d, n)$ は d 正則であるから, この場合 loop-rooted tree により因子分解を行なうことは不可能である。

次に, $g = 1$ と仮定する。このとき, $G_B(d, n)$ が loop-rooted tree で因子分解可能である

ことを示す． $A(G_B(d, n))$ を以下で表す d 個の部分弧集合 A_0, A_1, \dots, A_{d-1} に分解する．

$$\begin{aligned} A_0 &= \{e(v; k) \mid v = 0, 1, \dots, x_1 - 1, k = 0, 1, \dots, d - 1\} \\ &\cup \{e(x_1; k_1) \mid k_1 = 0, 1, \dots, q_1 - 1\} \end{aligned}$$

$$\begin{aligned} A_m &= \{e(x_m; k'_m) \mid k'_m = q_m, \dots, d - 1\} \\ &\cup \{e(v; k) \mid v = x_m + 1, \dots, x_{m+1} - 1, k = 0, 1, \dots, d - 1\} \\ &\cup \{e(x_{m+1}; k_{m+1}) \mid k_{m+1} = 0, 1, \dots, q_{m+1} - 1\}, 0 < m < d - 1 \end{aligned}$$

$$\begin{aligned} A_{d-1} &= \{e(x_{d-1}; k'_{d-1}) \mid k'_{d-1} = q_{d-1}, \dots, d - 1\} \\ &\cup \{e(v; k) \mid v = x_{d-1} + 1, \dots, n - 1, k = 0, 1, \dots, d - 1\}. \end{aligned}$$

補題 4.28 より, 各 A_i の弧の接続先は 0 から $n - 1$ までの連続する n 個の値であるから, $\langle A_i \rangle$ は $G_B(d, n)$ の 1 入正則な全域誘導部分ダイグラフである．これらの全域誘導部分ダイグラフが全域な loop-rooted tree であることを示す． A_i ($0 \leq i \leq d - 1$) に含まれるループの k -value を j とおくと, l_i は自分より大きな値を持つ $\min\{d - (j + 1), l_i\}$ 個の連続した値を持つ頂点に隣接し, 自分より小さな値を持つ $\min\{j, l_i\}$ 個の連続した頂点に隣接している．これらの隣接頂点のうち, 最大の値を持つものを t_1 , 最小の値を持つものを b_1 とおくと, $l_i + 1$ は $t_1 + 1$ から連続して $\max\{t_1 + d, n - 1\}$ までの頂点に隣接し, $l_i - 1$ は $b_1 - 1$ から連続して $\min\{b_1 - d, 0\}$ までの値の頂点に隣接する．この操作を $\min\{b_y, 0\} = 0$, $\max\{t_z, n - 1\} = n - 1$ となるまで繰り返すことで l_i から $G_B(d, n)$ のすべての頂点への有向パスが与えられ, $\langle A_i \rangle$ が loop-rooted tree であることが示せた． \square

$G_K(d, n)$ においては $h = \gcd(n + 1, d) = 1$ のとき, 定理 4.26 によりループの総数は d 個である． $G_K(d, n)$ についても同様の方法で示すことができる． $G_B(d, n)$ と同様に d 個のループ頂点を l_0, l_1, \dots, l_{d-1} ($l_0 < l_1 < \dots < l_{d-1}$) とおき, $G_B(d, n)$ 上で 0 へ隣接する頂点を x_0, x_1, \dots, x_{d-1} , ($x_0 < x_1 < \dots < x_{d-1}$) とおくと, 次が成り立つ．

補題 4.30 $x_{d-1} = n - 1$ であり, i ($0 \leq i < d - 1$) に対し $l_i \leq x_i \leq l_{i+1}$ を満たす．

証明 $-d(n - 1 + 1) \equiv 0 \pmod{n}$ であるから, $x_{d-1} = n - 1$ である． i ($0 \leq i < d - 1$) に対し $l_i \leq x_i \leq l_{i+1}$ であることを示す．まず, 0 から x_0 までのどの頂点もループを持たないと仮定する．今, 与えられた範囲内の頂点 v が自身以下の値を持つ頂点へ隣接しているとすると, 0 が $n - 1$ と隣接しており, 範囲内の頂点は連続であるから v までのいずれかの頂点はループを持ち, 矛盾が生じる．このことから範囲内のどの頂点に対しても自身以下の値を持つ頂点へ隣接することがない．しかしながら x_{i+1} は 0 へ隣接しており, 各頂点の値は 0 以上であるから必ずどこかにループを持つ頂点が少なくとも一つ存在する．同様に, $j > 0$ に

対しても x_j または $x_j + 1$ は $n - 1$ へ隣接し x_{j+1} は 0 へ隣接することから，その間にループを持つ頂点がただ一つ存在する．ループ頂点の総数を考えると，鳩の巣原理よりこのようなループを持つ頂点は 0 から x_0 の間及び x_i と x_{i+1} の間にただ一つ存在する． \square

図 4.5 に上記の定理より得られる loop-rooted tree による $G_B(4, 23)$ の因子分解を示す．

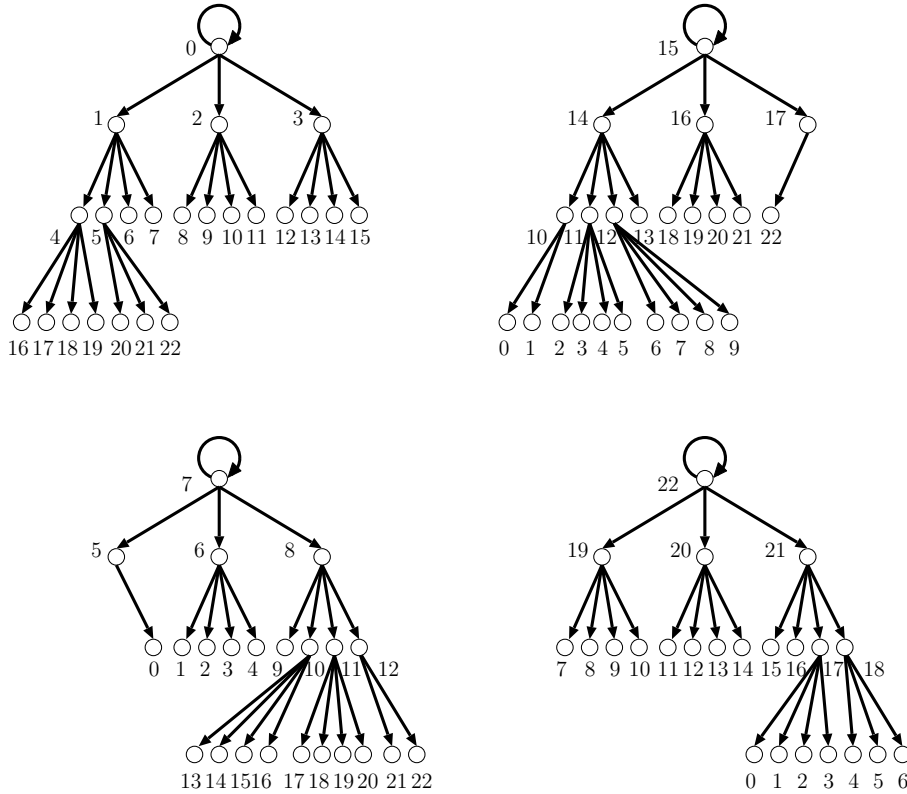


図 4.5: loop-rooted tree による $G_B(4, 23)$ の因子分解

p_i, q_i を $G_B(d, n)$ と同様に定義すると，以下が得られる．

定理 4.31 d, n を $d < n$ を満たす二つの自然数とし， $h = \gcd(d + 1, n)$ とする．このとき， $G_K(d, n)$ が loop-rooted tree により因子分解可能であるための必要十分条件は $h = 1$ であることである．

証明 $h > 1$ と仮定する．このとき，定理 4.27 より $G_K(d, n)$ のループの総数が d 個となることはない． $G_K(d, n)$ は d 正則であるから，この場合 loop-rooted tree により因子分解を行なうことは不可能である．

次に， $h = 1$ と仮定する．このとき， $G_K(d, n)$ が loop-rooted tree で因子分解可能である

ことを示す． $A(G_K(d, n))$ を以下で表す d 個の部分弧集合 A_0, A_1, \dots, A_{d-1} に分解する．

$$\begin{aligned} A_0 &= \{e(v; k) \mid v = 0, 1, \dots, x_0, k = 0, 1, \dots, d-1\} \\ &\cup \{e(x_0; k_0) \mid k_0 = q_0, q_0 + 1, \dots, d-1\}, \\ \\ A_m &= \{e(x_{m-1}; k_{m-1}) \mid k_{m-1} = 0, 1, \dots, q_{m-1} - 1\} \\ &\cup \{e(v; k) \mid v = x_{m-1} + 1, \dots, x_m, k = 0, 1, \dots, d-1\} \\ &\cup \{e(x_m; k_m) \mid k_m = q_m, q_m + 1, \dots, d-1\}, 1 \leq i \leq d-1. \end{aligned}$$

補題 4.30 より，各 $\langle A_i \rangle$ は 1 入正則な全域誘導部分ダイグラフである．これらの全域誘導部分ダイグラフにループ頂点から任意の頂点への有向パスが存在することを示すことにより，各 A_i が全域な loop-rooted tree であることを示す． A_i ($0 \leq i \leq d-1$) に含まれるループの k -value を j とおくと， l_i は自分より大きな値を持つ $\min\{d - (j + 1), l_i\}$ 個の連続した値を持つ頂点に隣接し，自分より小さな値を持つ $\min\{j, l_i\}$ 個の連続した頂点に隣接している．これらの隣接頂点のうち，最大の値を持つものを t_1 ，最小の値を持つものを b_1 とおくと， $t_1 \neq l_i$ のとき $l_i + 1$ は $b_1 - 1$ から連続して $\min\{b_1 - d, 0\}$ までの値の頂点に隣接し，同様に $b_1 \neq l_i$ のとき $l_i + 1$ は $t_1 + 1$ から連続して $\max\{t_1 + d, n - 1\}$ までの頂点に隣接する．この操作を $\min\{b_y, 0\} = 0, \max\{t_z, n - 1\} = n - 1$ となるまで繰り返すことで， l_i から $G_I(d, n)$ のすべての頂点への有向パスが与えられ， $\langle A_i \rangle$ が loop-rooted tree の定義を満たす． \square

4.3.2 loop-rooted tree による同型因子分解

この小節では特別なパラメータを持つ一般化 de Bruijn ダイグラフ $G_B(d, n)$ に対し，loop-rooted tree で同型因子分解を行なうための方法を示す．特に，可能な限り大きな完全 d 進 loop-rooted tree が構成可能なための条件を与え，いくつかの場合において実際に同型因子分解の方法を与える．

$d^p \leq n < d^{p+1}$ であるような d と n に対し， $G_B(d, n)$ を考えると，その中で構成可能な完全 d 進 loop-rooted tree は高さが高々 p のものである．次の定理で高さ p の完全 d 進 loop-rooted tree が構成できるような d と n の関係について示す．

定理 4.32 d, n を $d \geq 2$ かつ $d^p \leq n < d^{p+1}$ および， $d-1 \mid n-1$ を満たす整数とする．このとき， $G_B(d, n)$ は各ループ頂点を root-cycle とする高さ p の完全 d 進 loop-rooted tree を部分グラフとして含む．

証明 $d-1 \mid d^p - 1$ であり，条件より $d-1 \mid n-1$ であるから $n = d^p + a(d-1)$ とおける．さらに， $d^p \leq n < d^{p+1}$ であるから $0 \leq a < d^p$ を満足している．今，定理 4.25 より，条件

を満たす $G_B(d, n)$ の d 個のループ頂点 l_x ($x = 0, 1, \dots, d-1$) は

$$\begin{aligned} l_x &= x \frac{n-1}{d-1} = x \left(\frac{d^p + a(d-1)}{d-1} \right) \\ &= x \sum_{i=0}^{p-1} d^i + ax, \quad x = 0, 1, \dots, d-1, \end{aligned}$$

と表せる．各 l_x を root-cycle とする loop-rooted tree が存在することを示す． $G_B(d, n)$ に対し，以下の頂点部分集合を考える．

$$\begin{aligned} V_0 &= \{l_x\}; \\ V_1^- &= \{l_x + k \mid -x \leq k \leq -1\}; \\ V_1^+ &= \{l_x + k \mid 1 \leq k \leq d-1-x\}; \\ V_m^- &= \left\{ l_x + k \mid -x \sum_{i=0}^{m-1} d^i \leq k \leq -x \sum_{i=0}^{m-2} d^i - 1 \right\}; \\ V_m^+ &= \left\{ l_x + k \mid d^{m-1} - x \sum_{i=0}^{m-2} d^i \leq k \leq d^m - 1 - x \sum_{i=0}^{m-1} d^i \right\}, \text{ for } 2 \leq m \leq p. \end{aligned}$$

また， $V_i = V_i^- \cup V_i^+$ として定義する． $2 \leq m \leq p$ であるから，どの部分集合にも 0 より小さな値および $n = d^p + a(d-1)$ より大きな値を持つ頂点は存在しない．さらに V_m^- の最大要素は V_{m-1}^- の最小要素よりも 1 だけ値が小さく， V_m^+ の最小要素は V_{m-1}^+ の最大要素よりも 1 だけ値が大きいことから，二つの異なる部分集合間で n を法として合同な値は存在しない．加えて，上記のことからこれらのすべての部分集合の和集合は $l_x - x \sum_{i=0}^{m-1} d^i$ から $l_x + d^p - 1 - x \sum_{i=0}^{p-1} d^i$ までの連続する値を持つ d^p 個の頂点からなる．

$G_B(d, n)$ において l_x のループは x -value を持つので， $d \cdot l_x + x \equiv l_x \pmod{n}$ を満たす．したがって， l_x から隣接する d 個の頂点は

$$l_x - x + r \pmod{n}, \quad 0 \leq r \leq d-1,$$

で表される．よって l_x はループを持ち， V_1^- および V_1^+ の全ての頂点に隣接しており， $\langle V_0 \rangle$ は l_x を root-cycle とする高さ 1 の完全 loop-rooted tree である．

次に V_1^- について考える． V_1^- は $l_x - x$ から $l_x - 1$ までの連続する x 個の頂点からなる． $x(l_x - x; 0) = d(l_x - x) \equiv l_x - dx - x$ であるから $l_x - x$ は $l_x - dx - x$ から $l_x - dx - x + d - 1$ までの連続する d 個の頂点に隣接する．ここで，任意の頂点 w に対し $v(w; d-1) + 1 \equiv v(w+1; 0) \pmod{n}$ であることに注意すると V_1^- の頂点から隣接するすべての頂点の集合もまた連続であることがいえる． $|V_1^-| = x$ であるから隣接する頂点の集合は $l_x - dx - x$ から $l_x - x - 1$ までの連続する値を持つ dx 個の頂点からなり，これは V_2^- と一致する．同様に V_1^+ に対しても，最小の要素が $l_x + 1$ であり $|V_1^+| = d-1-x$ であるから， V_1^+ の頂点から隣接する頂点の集合は $l_x + d - x$ から $l_x + d^2 - dx - x - 1$ までの連続する $d^2 - d - dx$ 個の頂点から

なり, V_2^+ に一致する. よって $\langle V_0 \cup V_1 \rangle$ は頂点集合として $\bigcup_{i=0}^2 V_i$ を持ち, 弧集合として $\{e(v; r) \mid v \in V_0 \cup V_1, 0 \leq r \leq d-1\}$ を持つ高さ 2 の全域 loop-rooted tree となる.

$2 \leq m \leq p$ に対しても同様に考えると V_{m-1}^- の連続した値を持つ $d^{m-2}x$ 個の頂点は V_m^- の $d^{m-1}x$ 個の頂点に隣接先が重複しないように隣接し, V_{m-1}^+ の $d^{m-1} - d^{m-2} - d^{m-2}x$ 個の頂点についても V_m^+ の $d^m - d^{m-1} - d^{m-1}x$ 個のすべての頂点に重複なく隣接する. したがって $\langle \bigcup_{i=0}^{m-1} V_i \rangle$ は頂点集合として $\bigcup_{i=0}^m V_i$ を持ち, 弧集合として $\{e(v; r) \mid v \in \bigcup_{i=0}^{m-1} V_i, 0 \leq r \leq d-1\}$ を持つ高さ p の全域 loop-rooted tree となる. \square

l_x のループを root-cycle とする高さ m の完全 d 進 cycle-rooted tree を $x-CRT_m$ と表す. このとき,

$$\begin{aligned} V(x-CRT_m) &= \bigcup_{i=0}^m V_x^i = \left\{ l_x + k \mid -x \sum_{i=0}^{m-1} d^i \leq k \leq d^m - 1 - x \sum_{i=0}^{m-1} d^i \right\}, \\ A(x-CRT_m) &= \left\{ e(v; k) \mid v \in \bigcup_{i=0}^{m-1} V_x^i, k = 0, 1, \dots, d-1 \right\}, \end{aligned}$$

と表せる. これは, $x-CRT_m$ の弧集合が $\bigcup_{i=0}^{m-1} V_x^i$ に含まれる各頂点から出ているすべての弧のみからなることを示している. 二つの異なる $x-CRT_m$ に対して弧集合が持つ関係は次のようになる.

系 4.33 d, n を $d \geq 2$ かつ $d^p \leq n < d^{p+1}$ および, $d-1 \mid n-1$ を満たす整数とする. このとき, $G_B(d, n)$ の二つの異なるループ頂点 l_x, l_y に対し $(\bigcup_{i=0}^{p-1} V_x^i) \cap (\bigcup_{j=0}^{p-1} V_y^j) = \emptyset$ が成り立つ.

各 $x-CRT_p$ はそれぞれ互いに弧素である. また, $x-CRT_p$ は完全 d 進であるから, 根からの距離が $p-1$ 以下の頂点は全て d 個の異なる頂点への弧を持つ. そのため, どの $x-CRT_p$ にも含まれていない弧は, どの $x-CRT_{m-1}$ の頂点集合にも含まれていない頂点から出ている弧であると考えることができる. 以上のことから, どの $x-CRT_p$ にも含まれないような $G_B(d, n)$ の弧および隣接先の頂点を各 $x-CRT_p$ にそれぞれが全域かつ同型となるあるように付加することで, $G_B(d, n)$ を高さ p の loop-rooted tree を含む loop-rooted tree で同型因子分解可能であることが考えられる. この小節の残りでは実際に同型因子分解可能な $G_B(d, n)$ についての考察を行なう. 以降, d と n は $d-1 \mid n-1$ を満たす 2 以上の整数であり, a は $a \mid d$ を満たす整数として扱う.

$RV = V(G_B(d, n_g)) - \bigcup_{x=0}^{d-1} V(x-CRT_{n-1})$ を剰余頂点集合として定義し, $RA = \{e(v; k) \mid v \in RV, k = 0, 1, \dots, d-1\}$ を剰余弧集合として定義する. 定理 4.32 より, $d-1 \mid n-1$ のとき RV は次のように定式化できる.

$$RV = \{(i+1)d^{p-1} + ai + j \pmod{n} \mid i = 0, 1, \dots, d-2, j = 0, 1, \dots, a-1\}.$$

RV および RA を用いてそれぞれが同型であるような全域 cycle-rooted tree の構成を考えると、各 RV がどの頂点から隣接するかを把握することが重要となる。 RV の要素を以下のような $d-1$ 個の集合に分割する。

m ($0 \leq m \leq d-2$) に対し、

$$RV_m = \{ rv_{m,j} = (m+1)d^{p-1} + am + j \pmod{n} \mid j = 0, 1, \dots, a-1 \}, \\ m = 0, 1, \dots, d-2.$$

このとき、定理 4.32 の証明により、二つの異なる RV_m, RV'_m に属する要素はそれぞれ各 $x-CRT_p$ 内において異なる $d-1$ 個の高さ $p-1$ の部分木に含まれていることに注意する。

ここで、ラインダイグラフ演算を利用してダイグラフを分解するためにダイグラフ G の r -regular growth $\rho_r(G)$ を次のように定義する。

定義 4.34 (Hasunuma and shibata [30]) G を最大次数 Δ ($r \geq \Delta$) を持つダイグラフとする。 G の r -regular growth $\rho_r(G)$ とは各頂点 v に対し新たな $k = (r - \deg^+ v)$ 個の頂点 v_1, v_2, \dots, v_k と k 個の弧 $(v, v_1), (v, v_1), \dots, (v, v_k)$ を加えたダイグラフである。

この定義より、 G が cycle-rooted tree ならば $\rho_r(G)$ も同じ長さの root-cycle を持つ cycle-rooted tree であることがいえる。さらに、 $(L \cdot \rho_r)^n(G_k) = L(\rho_r(L \cdot \rho_r)^{n-1}(G))$ と再帰的に定義することで r -regular growth に対し以下が示されている。

定理 4.35 (Hasunuma and Shibata [30], Corollary 2.4) G を d 正則なダイグラフとする。 G が G_1, G_2, \dots, G_m により分解されるならば、 $L^k(G)$ は $(L \cdot \rho_r)^k(G_1), (L \cdot \rho_r)^k(G_2), \dots, (L \cdot \rho_r)^k(G_m)$ に分解される。

定理 4.35 より、 d 正則ダイグラフ G が同型因子分解可能であるならば G にラインダイグラフ演算を施して得られるダイグラフも同型因子分解可能であることがいえる。これを利用して $G_B(d, n)$ の同型因子分解を与える。

定理 4.36 d, n を $d \geq 2$ かつ $n = d^q(d^p + a(d-1))$, ($p \geq 2, q \geq 0$) を満たす整数とする。このとき、 $a|d$ かつ $a \leq d^{p-2}$ を満たすならば $G_B(d, n)$ は高さ $q+p+1$ の loop-rooted tree に同型因子分解できる。

証明 $a|d$ であるから、ある整数 d' に対し $d = ad'$ とおくことができる。まず $n = d^p + a(d-1)$ の場合を考える。系 4.33 より、それぞれが互いに弧素であるような d 個の $x-CRT_p$ が存在する。

$$V(x-CRT_p) = \bigcup_{i=0}^{m-1} V_x^i = \{ax, ax+1, \dots, d^p-1+ax\},$$

であり, $0 \leq x \leq d-1$ であるから, 与えられた a の条件の下で任意の x に対し各 $x\text{-}CRT_p$ は RV の全ての要素を含む. さらに RV の定義より, これらは全て各 $x\text{-}CRT_p$ の葉である. $d \cdot rv_{m,j} \equiv d \cdot rv_{m+1,j} + a \pmod{n}$ であることに注目すると,

$$v(rv_{m,j}; z) + 1 \equiv v(rv_{m+1,j}; z - a) \pmod{n}, \quad a \leq z \leq d-1,$$

が成り立つ. ここで d 個の頂点部分集合として,

$$\begin{aligned} V[d'j + ak] &= \{v(rv_{m,j}; f) \mid 0 \leq m \leq d-2, ak \leq f \leq a(k+1) - 1\}, \\ j &= 0, 1, \dots, a-1, \quad k = 0, 1, \dots, d' - 1, \end{aligned}$$

を定義する. このとき, どの部分集合の間にも $m = m', j = j'$ かつ $f = f'$ を満たすような二つの要素 $v(rv_{m,j}; f), v(rv_{m',j'}; f')$ は存在しない. したがって各部分集合に含まれる頂点 $v(rv_{m,j}; f)$ は隣接先の頂点 $rv_{m,j}$ と, その頂点から出ている弧 $e(rv_{m,j}; f)$ は一意に決まる. さらに, 各部分集合は $v(rv_{0,j}; ak)$ から連続する $a(d-1)$ 個の頂点を持ち,

$$\begin{aligned} v(rv_{0,j}; ak) &= d \cdot rv_{0,j} + ak \\ &= d(d^{p-1} + j) + ak \\ &\equiv d^p + dj + ak \pmod{n}, \end{aligned}$$

であるから, 各 $V[d'j + ak]$ は,

$$V[d'j + ak] = \{d^p + dj + ak, d^p + dj + ak + 1, \dots, d^p + dj + ak + a(d-1) - 1 \pmod{n}\},$$

と表すことができ, これらの値は n を法として異なる. 一方で, 各 $x\text{-}CRT_p$ に対し

$$\begin{aligned} V(G_B(d, n)) - V(x\text{-}CRT_p) &= \{0, 1, \dots, ax - 1, d^p + ax, d^p + ax + 1, \dots, n - 1\} \\ &= \{d^p + ax, d^p + ax + 1, \dots, d^p + ax + a(d-1) - 1 \pmod{n}\}, \end{aligned}$$

であるので, $j = \lfloor \frac{x}{d'} \rfloor, k = x \pmod{d'}$ のとき $V(G_B(d, n)) - V(x\text{-}CRT_p) = V[d'j + ak] = V[x]$ が成り立つ. したがって d 個の $V(G_B(d, n)) - V(x\text{-}CRT_p)$ と $V[d'j + ak]$ が一対一で対応していることがいえる. 以上のことから $V[x]$ の各要素に対応する弧集合として

$$A[x] = \{e(rv_{m,j}; f) \mid v(rv_{m,j}; f) \in V[x]\},$$

を定義すると, $V(x\text{-}CRT_p) \cup \langle V[x] \rangle$ は $G_B(d, n)$ の全域 cycle-rooted tree $SPCRT_x$ となること が得られる. さらに, $SPCRT_x$ 内において $V(x\text{-}CRT_p)$ の $d-1$ 個の葉 $rv_{0,j}, rv_{1,j}, \dots, rv_{d-2,j}$ から $V[x] = V[d'j + ak]$ の頂点にそれぞれ a 本ずつ接続先が異なる弧が出ている. 今, $V(x\text{-}CRT_p)$ において $rv_{0,j}, rv_{1,j}, \dots, rv_{d-2,j}$ はそれぞれ異なる高さ $p-1$ の部分木に属しているため, 各 $SPCRT_x$ は $x\text{-}CRT_p$ の高さ $p-1$ の各部分木のある一つの葉から a 本の弧が

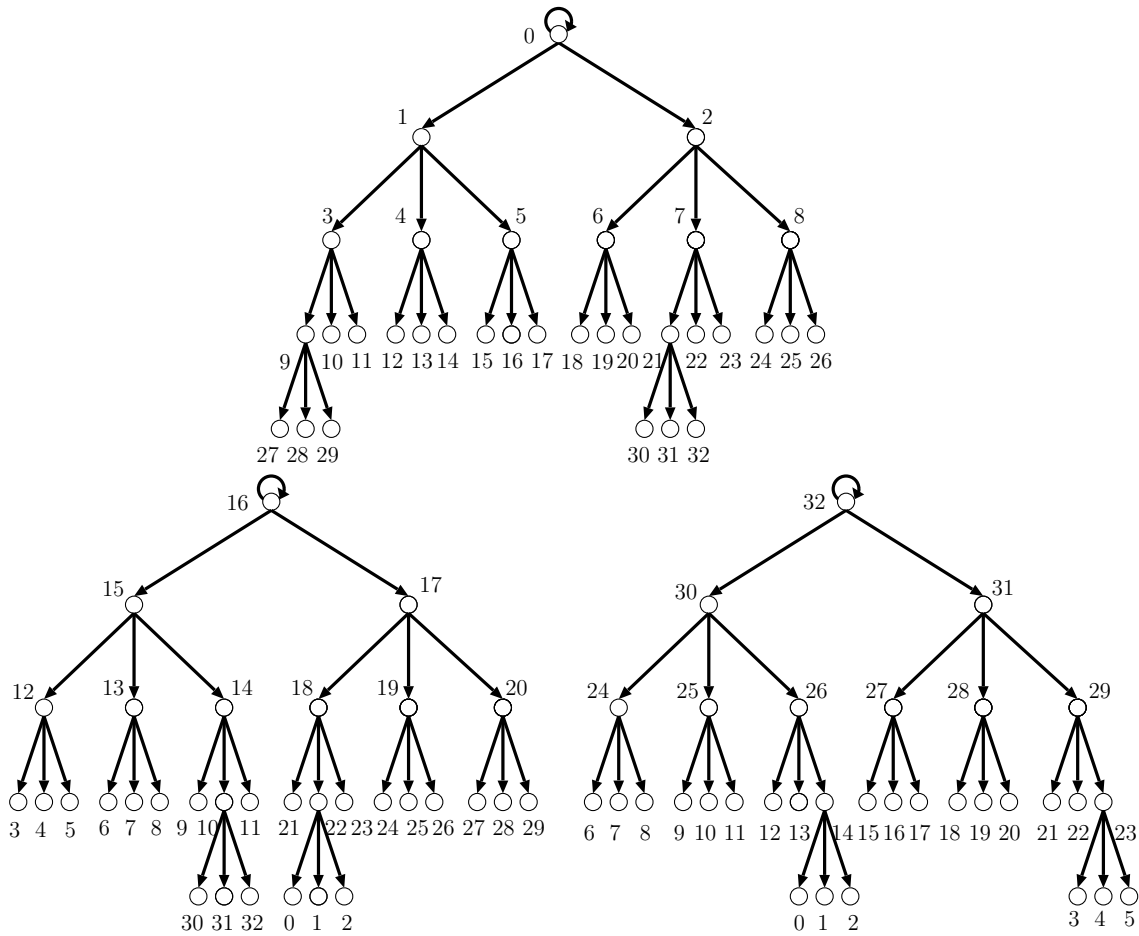


図 4.6: loop-rooted tree による $G_B(3, 33)$ の同型因子分解

出ている形をしている．したがって各 $SPCRT_x$ は同型であり， $G_B(d, n = d^p + a(d - 1))$ が loop-rooted tree で同型因子分解可能であることが示せた．

最後に， $L^k(G_B(d, n)) = G_B(d, d^k n)$ であるから定理 4.35 より， $L^k(G_B(d, n))$ は $(L \cdot \rho_r)^k(SPCRT_0), (L \cdot \rho_r)^k(SPCRT_1), \dots, (L \cdot \rho_r)^k(SPCRT_{d-1})$ で同型因子分解可能であり，各 $(L \cdot \rho_r)^k(SPCRT_x)$ もまた loop-rooted tree であるから命題は成り立つ． \square

図 4.6 に上記の定理から得られる loop-rooted tree による $G_B(3, 33)$ の同型因子分解を示す．

4.3.3 複数のループを同時に含む同型因子分解

本小節では $g > 1$ の場合の $G_B(d, n)$ ，および $h = 1$ の場合の $G_K(d, n)$ を同型因子分解することを考える． $g, h > 1$ の場合については loop-rooted tree によって同型因子分解することは不可能であることはすでに述べた．また，ループを持つ頂点は同じくループを持つ頂点

への同型写像しか存在しないため, $G_B(d, n), G_K(d, n)$ を同型因子分解したときに得られる弧の分解集合の数はそれぞれのループ数の約数である. したがって, これらのダイグラフを同型因子分解するためには弧の各分割集合の中に同数のループが含まれている必要がある.

同型因子分解を行なうに際し, $G_B(d, n)$ の自己同型写像として次を与える.

定理 4.37 d, n を $d \geq 2, n \geq 1$ を満たす整数とし, $g = \gcd(d-1, n)$, $d-1 = gs_g$, $n = gt_g$ とおく. このとき, $\phi_i(v) = v + it_g \pmod{n}$, $0 \leq i \leq g-1$ は $G_B(d, n)$ の自己同型写像である.

証明 $G_B(d, n)$ の任意の頂点 v に対し, $\phi_i(v - it_g) = v$ を満たす $G_B(d, n)$ の頂点 $v - it_g \pmod{n}$ が必ず存在する. したがって ϕ_i は surjection である. また, 二つの異なる値 v と v' に対して $\phi_i(v) \equiv \phi_i(v') \pmod{n}$ が成り立つと仮定すると, $v + it_g \equiv v' + it_g \pmod{n}$ であるから $v \equiv v' \pmod{n}$ となり, n を法として v と v' の値は等しいので v と v' は $G_B(d, n)$ 上では同一の頂点である. よって $\phi_i(v)$ は injection である. 以上より $\phi_i(v)$ は bijection である. 次に ϕ_i が $V(G_B(d, n))$ 上での隣接関係を保存することを示す. 任意の $v \in V(G_B(d, n))$ に対し v から隣接している頂点は $dv + r \pmod{n}$, ($r = 0, 1, \dots, d-1$) である. 同様にして, $\phi_i(v) = v + it_g \pmod{n}$ から隣接している頂点は

$$dv + dit_g + r = dv + (d-1)it_g + it_g + r,$$

であり, $d-1 = gs_g$ なので $(d-1)it_g = gs_g it_g = s_g in$ となり

$$\begin{aligned} &\equiv dv + it + r \pmod{n} \\ &= \phi_i(dv + r), \end{aligned}$$

であるから ϕ_i は隣接関係を保っている. したがって, ϕ_i は $V(G_B(d, n))$ 上で自己同型写像をなす. □

同様に, $G_K(d, n)$ に対しても次のような自己同型写像が存在する.

定理 4.38 d, n を $d \geq 2, n \geq 1$ を満たす整数とし $h = \gcd(d+1, n)$, $d+1 = hs_h$, $n = ht_h$ とおく. このとき $\phi_i(v) = v + it_h \pmod{n}$, $0 \leq i \leq h-1$ は $G_K(d, n)$ の自己同型写像である.

次に, これらの自己同型写像を用いることで $G_B(d, n), G_K(d, n)$ がそれぞれ g 個および h 個の同型な因子に分解できることを示す. 定義 4.37 の結果を用いて $g \geq 2$ における $G_B(d, n)$ の同型因子分解を与える.

定理 4.39 d, n を $d \geq 2, n \geq 1$ を満たす整数とし, $g = \gcd(d-1, n)$, $d+1 = gs_g$, $n = gt_g$ とおく. また, $A_i = \{e(v; r) | v = it_g, it_g + 1, \dots, (i+1)t_g - 1, r = 0, 1, \dots, d-1\}$, ($0 \leq i \leq g-1$) と定める. このとき $G_B(d, n)$ は $\langle A_i \rangle$ によって同型因子分解される.

証明 定理 4.25 より, 同じ k -value を持つループ頂点の値はそれぞれ少なくとも t 以上離れている. 一方, $n = gt_g$ であるから各々異なる $\langle A_i \rangle$ に含まれていることがいえる. よって $\langle A_i \rangle$ に含まれるループ頂点は $s_g - 1$ 個である. 今, 各 $|\langle A_i \rangle| = t_g$ であり $\langle A_i \rangle$ に含まれる要素は連続した頂点に隣接しているから, $dt_g = (d+1)t_g - t_g = s_g n - t_g$ となる. さらに, $1 < s_g < d+1, 1 < t_g < n$ であるから $s_g n - t_g > n$ を満たす. よって $\langle A_i \rangle$ は $G_B(d, n)$ の因子分解となっている.

ここで, $V(G_B(d, n))$ から $V(G_B(d, n))$ への写像 $\phi_{ij}(v) = v + (j-i)t_g \pmod{n}$ を考えると, 定理 4.37 よりこれは $V(G_B(d, n))$ 上での自己同型写像になっている. $V(G_B(d, n)) = \langle A_i \rangle$ であるから, ϕ_{ij} は $\langle A_i \rangle$ から $\langle A_j \rangle$ への写像としても用いることができる. さらに, $\langle A_i \rangle$ の頂点 $it_g + k$ ($0 \leq k < t_g$) は $\langle A_j \rangle$ の頂点 $jt_g + k$ に写り $d(it_g + k) + r$ は $d(jt_g + k) + r$ へ写るため, この場合においても ϕ_{ij} は隣接関係及び非隣接関係を保存している. したがって, ϕ_{ij} は $\langle A_i \rangle$ から $\langle A_j \rangle$ への同型写像である. \square

$G_K(d, n)$ についても同様に以下が成り立つ.

定理 4.40 d, n を $d \geq 2, n \geq 1$ を満たす整数とし, $h = \gcd(d+1, n)$, $d+1 = hs_h$, $n = ht_h$ とおく. また, $A_i = \{(v, -d(v+1)+r) | v = it_h, it_h+1, \dots, (i+1)t_h-1, r = 0, 1, \dots, d-1\}$, ($0 \leq i \leq h-1$) と定める. このとき, $1 < h < d+1$ ならば $G_K(d, n)$ は $\langle A_i \rangle$ によって同型因子分解される.

図 4.7 に $G_B(4, 21)$ の同型因子分解の例を示す.

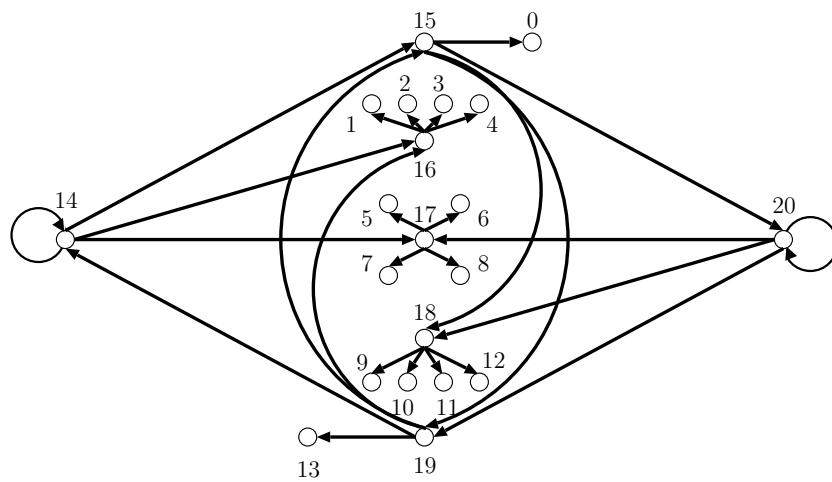
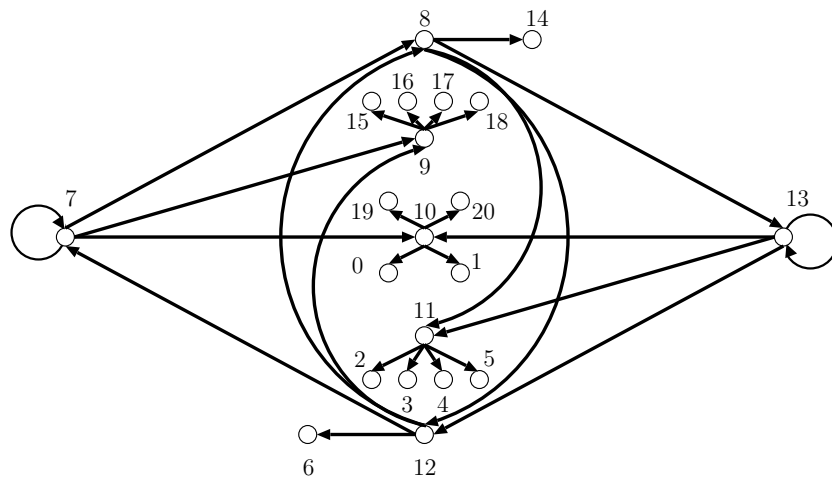
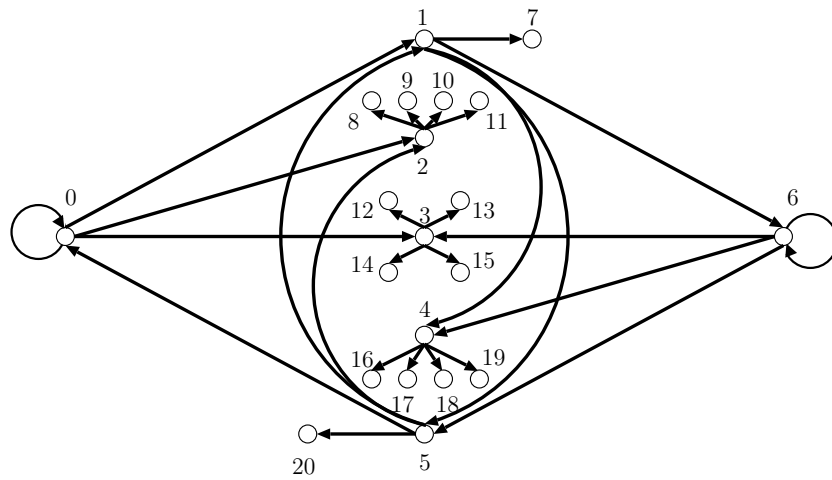


図 4.7: $G_B(4, 21)$ の同型因子分解

第5章 de Bruijn ダイグラフ間の包含関係

de Bruijn ダイグラフ $B(d+1, n)$ は Kautz ダイグラフ $K(d, n)$ を誘導部分ダイグラフとして含んでいることが定義よりわかる．また， $B(d, n)$ は $B(d-1, n)$ を誘導部分ダイグラフとして含み，Kautz ダイグラフ $K(d, n)$ は $K(d, n-1)$ を誘導部分ダイグラフとして含む．これらの関係をもとに，本章では de Bruijn 族のグラフおよびその積で構成されるグラフに対する相互関係について述べる．

5.1 準同型写像を用いた de Bruijn ダイグラフの分解

グラフの準同型写像とは隣接関係を保存するような，大きなネットワークから小さなネットワークへの頂点の写像であり，グラフ理論を用いて小さなネットワーク上で大きな並列計算をシミュレートするための方法の一つである．

準同型写像を用いたシミュレーションの方法は Fishburn と Finkel が導入し，グラフの emulation として一般化され [22]，パスやリング，ハイパーキューブ，Cayley ダイグラフなどのグラフに対しての結果が得られている [2, 8, 9]．

de Bruijn ダイグラフに対して， $B(d, n)$ から $B(d, n-1)$ への頂点一様，および弧一様であるような準同型写像についての考察が [60] に示されている．再帰的にこのような準同型写像を与えることによって， $V(B(d, n))$ から $V(B(d, 1))$ ，すなわち K_d^* への準同型写像を与えることができる．ここでは $B(d, n)$ の頂点の和および差による演算で定義される準同型写像について注目し， K_d^* へ写像された結果から得られる誘導部分ダイグラフについての考察を行う．

和による準同型写像 $\phi_a : V(B(d, n)) \rightarrow V(B(d, n-1))$ を次のように定義する．

$$\phi_a(x_0, x_1, \dots, x_{n-1}) = ((x_0 \oplus_d x_1), (x_1 \oplus_d x_2), \dots, (x_{n-2} \oplus_d x_{n-1})),$$

ここで， \oplus_d は d を法とする和を表す．同様に，差による準同型写像 $\phi_s : V(B(d, n)) \rightarrow V(B(d, n-1))$ を次で定義する．

$$\phi_s(x_0, x_1, \dots, x_{n-1}) = ((x_0 \ominus_d x_1), (x_1 \ominus_d x_2), \dots, (x_{n-2} \ominus_d x_{n-1})),$$

\ominus_d は d を法とする差を表す．これらの準同型写像を繰り返し適用することで， $V(B(d, n))$ から $V(B(d, n-m))$ ($1 \leq m \leq n-1$) への写像が得られる．これを m -反復準同型写像と呼

び, ϕ^m と表す.

ϕ_a, ϕ_s の一様性が, [60] で示されている.

定理 5.1 (Tvrđik, Harbane and Heydemann [60], Corollary 13) $V(B(d,n))$ から $V(B(d,n-1))$ への準同型写像 ϕ_a, ϕ_s は頂点一様かつ弧一様である.

$|V(B(d,n))| = d^n, |V(B(d,n-1))| = d^{n-1}$ であるから, 式 (2.1) を用いると ϕ_a, ϕ_s において $B(d,n-1)$ の各頂点はちょうど d 個の $B(d,n)$ の頂点から写像されることがわかる. 同様に, 式 (2.2) により $B(d,n-1)$ の各弧がちょうど d 本の $B(d,n)$ の弧から写像されていると考えられる.

5.1.1 m -反復準同型写像によるダイグラフ

この小節では $V(B(d,n))$ の各頂点に対し, m -反復準同型写像によって写像された K_d^* 上での値をもとに頂点集合を分割することを考える. 和と差による準同型写像に対し, 以下の基本的な性質が成り立つ.

定理 5.2 $V(B(d,n))$ から $V(K_d^*)$ への $(n-1)$ -反復準同型写像 $\phi_a^{n-1}, \phi_s^{n-1}$ に対し,

$$\phi_a^{n-1}(x_0 \oplus_d \alpha, x_1, \dots, x_{n-1}) = \alpha \oplus_d (\phi_a^{n-1}(x_0, x_1, \dots, x_{n-1})) \quad (5.1)$$

$$\phi_a^{n-1}(x_0, x_1, \dots, x_{n-1} \oplus_d \alpha) = \alpha \oplus_d (\phi_a^{n-1}(x_0, x_1, \dots, x_{n-1})) \quad (5.2)$$

$$\phi_s^{n-1}(x_0 \oplus_d \alpha, x_1, \dots, x_{n-1}) = \alpha \oplus_d (\phi_s^{n-1}(x_0, x_1, \dots, x_{n-1})) \quad (5.3)$$

$$\phi_s^{n-1}(x_0, x_1, \dots, x_{n-1} \oplus_d \alpha) = \alpha \oplus_d (\phi_s^{n-1}(x_0, x_1, \dots, x_{n-1})). \quad (5.4)$$

ただし, α は $d-1$ 以下の任意の正整数である.

和および差による準同型写像において, 先頭および末尾のアルファベットは他の桁の演算に対し影響を与えないことから, 定理 5.2 は容易に得られる. この事実から, さらに次が得られる.

定理 5.3 $\alpha \neq \beta$ であるような任意の $\alpha, \beta \in \mathbb{Z}_d$ に対し,

$$\phi_a^{n-1}(\alpha, x_1, \dots, x_{n-1}) \neq \phi_a^{n-1}(\beta, x_1, \dots, x_{n-1}) \quad (5.5)$$

$$\phi_a^{n-1}(x_0, \dots, x_{n-2}, \alpha) \neq \phi_a^{n-1}(x_0, \dots, x_{n-2}, \beta) \quad (5.6)$$

$$\phi_s^{n-1}(\alpha, x_1, \dots, x_{n-1}) \neq \phi_s^{n-1}(\beta, x_1, \dots, x_{n-1}) \quad (5.7)$$

$$\phi_s^{n-1}(x_0, \dots, x_{n-2}, \alpha) \neq \phi_s^{n-1}(x_0, \dots, x_{n-2}, \beta). \quad (5.8)$$

また，和による準同型写像と差による準同型写像が持つ異なる性質として，次が与えられる．

定理 5.4 $B(d, n)$ の任意の頂点 $v = (v_0, v_1, \dots, v_{n-1})$ とおき， $v \oplus_d \alpha = (v_j \oplus_d \alpha)$ ， $0 \leq j \leq n-1$ と定義する．このとき，

$$\begin{cases} \phi_a^{n-1}(v) \neq \phi_a^{n-1}(v \oplus_d \alpha) & \text{if } n \text{ is odd,} \\ \phi_a^{n-1}(v) = (\phi_a^{n-1}(v \oplus_d \alpha)) \oplus_d 2k, \exists k & \text{if } n \text{ is even,} \end{cases} \quad (5.9)$$

$$\phi_s^{n-1}(v) = \phi_s^{n-1}(v \oplus_d \alpha). \quad (5.10)$$

$V(B(d, n))$ から $V(K_d^*)$ への $(n-1)$ -反復準同型写像 $\phi_a^{n-1}, \phi_s^{n-1}$ によって写像される頂点の値によって $B(d, n)$ の頂点を分割することを考える．頂点の分割集合を次のように定義する．

$$PV_i = \{v \mid v \in B(d, n) \text{ and } \phi_a^{n-1}(v) = i\}, \quad (5.11)$$

$$MV_i = \{v \mid v \in B(d, n) \text{ and } \phi_s^{n-1}(v) = i\}, \quad 0 \leq i \leq d-1. \quad (5.12)$$

上記の定義より， $PV_i(MV_i)$ は ϕ_a^{n-1} により K_d^* の頂点 i に写像される $B(d, n)$ のすべての頂点を含む．今， PV_i および MV_i の頂点により誘導される部分ダイグラフについて，以下が得られる．

定理 5.5 $B(d, n)$ の頂点部分集合 PV_i によって誘導される部分ダイグラフを $\langle PV_i \rangle$ とおく．このとき， $\langle PV_i \rangle$ は有向サイクルの和集合である．

証明 PV_i が 1 正則ダイグラフであることを示す． $B(d, n)$ 上の頂点 $v = (v_0, v_1, \dots, v_{n-1})$ に対し， $\phi_a^{n-1}(v) = i$ であると仮定する．このとき，定理 5.3-(5.6) より， $(v_1, v_2, \dots, v_{n-1}, \alpha)$ と表される頂点はただ一つだけ PV_i に含まれている．したがって， $\langle PV_i \rangle$ 上における v の出字数は 1 となる．同様に，定理 5.3-(5.5) より $(\beta, v_0, v_1, \dots, v_{n-2})$ と表される頂点はただ一つだけ PV_i に含まれており， v の入字数は 1 となる．以上より， $\langle PV_i \rangle$ は 1 正則ダイグラフであり，有向サイクルの和集合である． \square

MV_i の誘導部分ダイグラフに対しても，定理 5.3-(5.7)(5.8) から同様に有向サイクルの和集合となる．

上記の定理により，各 PV_i, MV_i が有向サイクルの集合を誘導することがわかった．次に，各誘導部分ダイグラフがどのような有向サイクルを含んでいるかについての考察を行う．

定理 5.6 MV_i を $B(d, n)$ の頂点分割集合とおき， MV_i^- を $B(d, n-1)$ の頂点分割集合とおく．このとき，

$$\langle MV_0 \rangle \cong \bigcup_{0 \leq i \leq d-1} \langle MV_i^- \rangle$$

証明 $B(d, n-1)$ の各誘導部分ダイグラフ $\langle MV_i^- \rangle$ のラインダイグラフ $L(\langle MV_i^- \rangle)$ を考える . このとき , $\langle MV_i^- \rangle$ はそれぞれ有向サイクルの和集合であるから , $\langle MV_i^- \rangle \cong L(\langle MV_i^- \rangle)$ となる . 今 , $L(\langle MV_i^- \rangle)$ の各頂点を $B(d, n)$ 上の頂点として対応させて考える . MV_i^- 上で隣接する 2 頂点を $u = (x_0, x_1, \dots, x_{n-2}), v = (x_1, \dots, x_{n-2}, x_{n-1})$ とおくと , $\phi_s^{n-2}(u) = \phi_s^{n-2}(v) = i$ である . これら 2 頂点の弧から得られる $B(d, n)$ 上の頂点 $w = (x_0, x_1, \dots, x_{n-2}, x_{n-1})$ に対し , ϕ_s を再帰的に適用し $B(d, 1)$ で写像される値を見ていくと ,

$$\begin{aligned} \phi_s(w) &= ((x_0 \ominus_d x_1), (x_1 \ominus_d x_2), \dots, (x_{n-2} \ominus_d x_{n-1})) \\ \phi_s^2(w) &= ((x_0 \ominus_d x_1) \ominus_d (x_1 \ominus_d x_2), (x_1 \ominus_d x_2) \ominus_d (x_2 \ominus_d x_3), \dots, (x_{n-3} \ominus_d x_{n-2}) \ominus_d (x_{n-2} \ominus_d x_{n-1})) \\ &\vdots \\ \phi_s^{n-2}(w) &= (\phi_s^{n-2}(u), \phi_s^{n-2}(v)) \\ \phi_s^{n-1}(w) &= \phi_s^{n-2}(u) \ominus_d \phi_s^{n-2}(v) = i - i = 0, \end{aligned}$$

となり , すべての i に対し $L(\langle MV_i^- \rangle)$ に含まれる頂点は $\phi_s^{n-1}(w)$ によって $B(d, 1) \cong K_d^*$ の頂点 0 に写される . 次に , $B(d, n)$ のこれらの頂点以外が ϕ_s^{n-1} により $B(d, 1)$ の頂点 0 に写像されないことを示す . 今 , 上記によって与えられた , 頂点 0 に写像される $B(d, n)$ の頂点の総数は $\bigoplus_{0 \leq i \leq d-1} |L(\langle MV_i^- \rangle)| = d \times d^{n-2} = d^{n-1}$ である . 定理 2.1 により , ϕ_s は頂点一様な準同型写像であり , この反復準同型写像も頂点一様である . したがって , $B(d, n)$ に $(n-1)$ -反復準同型写像 ϕ_s^{n-1} を適用することで写像される $B(d, 1)$ の各頂点の負荷はそれぞれ d^{n-1} となる . これは上記の頂点の総数と等しいため , これらの頂点以外が頂点 0 に写像されることがないといえる . 以上から , $\bigcup_{0 \leq i \leq d-1} L(\langle MV_i^- \rangle) \cong \langle MV_0 \rangle$ と同型であり , $\bigcup_{0 \leq i \leq d-1} \langle MV_i^- \rangle \cong \langle MV_0 \rangle$ を導く . \square

和の準同型写像に対して次の考察が得られる .

定理 5.7 d を奇整数とし $B(d, n)$ の和による準同型写像を ϕ_a とおく . このとき , 任意の $i, j \in \mathbb{Z}_d$ に対し ,

$$\langle PM_i \rangle \cong \langle PM_j \rangle.$$

証明 定理 5.4-(5.9) 式により , $B(d, n)$ の頂点 $x = (x_0, x_1, \dots, x_{n-1})$ に対し , $x \oplus_d \alpha$, ($0 \leq \alpha \leq n-1$) は ϕ_a^{n-1} によりそれぞれ $B(d, 1)$ の異なる頂点へ写像される . 今 , PV_i のすべての頂点 x に対し , $x \oplus_d j$ となるような頂点集合 V_j を考える . $x \oplus_d j$ に ϕ_a^{n-1} を作用させると , 得られる値は $i \oplus_d j \cdot 2^n$ となる . 仮定より , d は奇であるから , これらの値はそれぞれ異なる \mathbb{Z}_d 内の値となる . したがって , V_j のすべての頂点は $B(d, 1)$ の頂点 $i \oplus_d j \cdot 2^n$ へと写像され , その誘導部分ダイグラフは $\langle PM_i \rangle$ と同型である . \square

図 5.1 に $B(3, 3)$ の和による準同型写像から構成された誘導部分ダイグラフを表す .

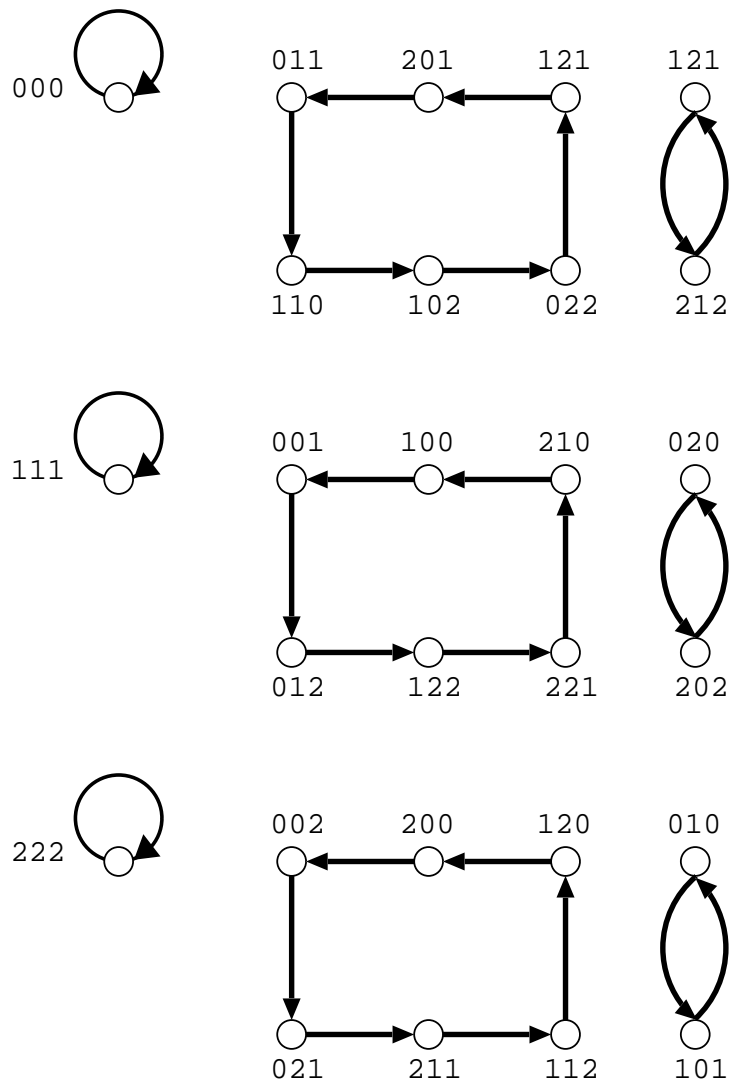


図 5.1: $B(3,3)$ の和による準同型写像から構成された誘導部分ダイグラフ

5.2 一般化 de Bruijn ダイグラフの包含関係

この節では de Bruijn 族グラフの積に対する同型因子分解について考察を行なう . de Bruijn 族グラフとラインダイグラフ , およびクロネッカー積の間には相互に密接な関係があり , ラインダイグラフとクロネッカー積の関係は [54] で述べられている .

[51] では同型因子分解という観点から de Bruijn ダイグラフとクロネッカー積をラインダイグラフを用いて考察し , de Bruijn ダイグラフによる同型因子分解を与えている . [40] では 2 進一般化 de Bruijn ダイグラフのクロネッカー積に対する分解による拡張した結果が得られている .

定理 5.8 (Kawai and Shibata [40], Theorem 4) $G_B(2, mn) \mid G_B(2, m) \otimes G_B(2, n)$.

さらに , 同論文では次のような包含関係が示されている .

定理 5.9 (Kawai and Shibata [40], Proposition 3) $G_B(d, mn) \subseteq G_B(d, m) \otimes G_B(d, n)$.

この証明において $G_B(d, mn)$ の頂点集合を $G_B(d, m) \otimes G_B(d, n)$ へ次のような関数を用いて写像している .

$$\sigma(x) = \left(\left[\frac{x}{n} \right], x \bmod n \right).$$

$H := \sigma(G_B(d, mn)) = (\sigma f_k \sigma^{-1} \mid k \in \mathbb{Z}_d)$ は写像元である $G_B(d, mn)$ と同型であり , これらが $G_B(d, m) \otimes G_B(d, n)$ の部分グラフであることが示されている ($f_k(x) = (dx + k) \bmod n$).

上記の包含関係をもとに , 一般化 de Bruijn ダイグラフのクロネッカー積が一般化 de Bruijn ダイグラフで同型因子分解可能な条件についての考察を行う . この結果は [40] の結果をさらに d 進について一般化したものとして捉えることができる .

5.2.1 ループを用いた同型因子分解のための必要条件

ダイグラフ G が ダイグラフ H により同型因子分解されるための必要条件として , G 内のループの総数 L_G が H 内のループの総数 L_H によって割り切られなければならない . 一般化 de Bruijn ダイグラフ $G_B(d, n)$ に対し , $g = \gcd(d-1, n)$ とおくと [29] よりループの総数は $d-1+g$ で与えられることがわかっている . G と H のクロネッカー積により得られるダイグラフ $G \otimes H$ において , ループは G 上のループ頂点 x と H 上のループ頂点 y からなる頂点 (x, y) 上のみ存在し , その総数は $L_G \times L_H$ である . 今 , $G_B(d, mn)$ が $G_B(d, m) \otimes G_B(d, n)$ を割り切るとき , $G_B(d, mn)$ は d 正則であり $G_B(d, m) \otimes G_B(d, n)$ は d^2 正則であるから , 分解の因子の個数は d 個である . したがって $G_B(d, mn) \mid G_B(d, m) \otimes G_B(d, n)$ を満たすためには $dL_{mn} = L_m L_n$ を満たす必要がある . 次の補題でこのための条件を示す .

補題 5.10 $g_m = \gcd(d-1, m), g_n = \gcd(d-1, n), g_{mn} = \gcd(d-1, mn)$ とおく . $dL_{mn} = L_m L_n$ を満たすための必要十分条件は以下の条件のどちらかを満たすことである .

- $g_m = 1$ または $g_n = 1$.
- d が奇数の 2 乗かつ $g_m = g_n = \sqrt{d} + 1$.

証明 $g_m = 1$ または $g_n = 1$ のどちらかを満たすとき , ここでは , $g_m = 1$ としたとき , $L_{mn} = L_n, L_m = d$ より条件を満たす . d が奇数の 2 乗かつ $g_m = g_n = \sqrt{d} + 1$ であるとき , $d-1 = (\sqrt{d}+1)(\sqrt{d}-1)$ より $\gcd(d-1, m) = \gcd(d-1, n) = \sqrt{d}+1$ である . よって , 与式は

$$\begin{aligned} d(d-1 + \gcd(d-1, (\sqrt{d}+1)^2)) &= (d + \sqrt{d})^2 \\ d^2 - d + d \cdot \gcd(d-1, d + 2\sqrt{d} + 1) &= (d + \sqrt{d})^2, \end{aligned}$$

となる . d は奇数の 2 乗であるから , $d = (2t-1)^2, t > 1$ とおくと

$$\begin{aligned} \gcd(d-1, d + 2\sqrt{d} + 1) &= \gcd((2t-1)^2 - 1, (2t-1)^2 + 2(2t-1) + 1) \\ &= \gcd(4t^2 - 4t, 4t^2) \\ &= 4t \gcd(t-1, t) \\ &= 4t, \end{aligned}$$

と表せる . さらに , $\sqrt{d} = 2t-1$ より $4t = 2\sqrt{d} + 2$ となる . これを上式に代入すると

$$\begin{aligned} d^2 - d + d(2\sqrt{d} + 2) &= (d + \sqrt{d})^2 \\ d^2 + 2d\sqrt{d} + d &= d^2 + 2d\sqrt{d} + d, \end{aligned}$$

となり等式を満たす .

逆に , 等式 $dL_{mn} = L_m L_n$ が成り立つと仮定する . このとき , 上式は

$$\begin{aligned} dL_{mn} &= L_m L_n \\ d(d-1 + g_{mn}) &= (d-1 + g_m)(d-1 + g_n), \end{aligned} \tag{5.13}$$

と表せる . 任意の整数 a, b, c に対し $\gcd(a, bc) = \gcd(a, \gcd(a, b) \gcd(a, c))$ であるから ,

$$g_{mn} = \gcd(d-1, g_m g_n), \tag{5.14}$$

となる . また , $\gcd(g_m, g_n) = r_{mn}$ とおき $g_m = g'_m r_{mn}, g_n = g'_n r_{mn}, d-1 = g'_n g'_m r_{mn} r_{d-1}$ とおくと , $\gcd(g'_m, g'_n) = 1$ であり $g_m g_n = g'_m g'_n r_{mn}^2$ であるから , $\gcd(d-1, g_m g_n) = g'_m g'_n r_{mn} \gcd(r_{mn}, r_{d-1})$ を得る . これらより , (5.13) 式は

$$\begin{aligned} d(d-1 + \gcd(d-1, g_m g_n)) &= (g'_n g'_m r_{mn} r_{d-1} + g'_m r_{mn})(g'_n g'_m r_{mn} r_{d-1} + g'_n r_{mn}) \\ &= g'_m r_{mn} g'_n r_{mn} (g'_n r_{d-1} + 1)(g'_m r_{d-1} + 1), \end{aligned}$$

となり, $g'_m r_{mn} = g_m, g'_n r_{mn} = g_n$ であるから,

$$d(d-1 + \gcd(d-1, g_m g_n)) = g_m g_n (g'_n r_{d-1} + 1)(g'_m r_{d-1} + 1) \quad (1'),$$

と表せる. ここで $d \perp g_m$ かつ $d \perp g_n$ であるから, 上式は

1. $d = (g'_n r_{d-1} + 1)(g'_m r_{d-1} + 1)$.
2. $d < (g'_n r_{d-1} + 1)(g'_m r_{d-1} + 1)$ かつ $d \mid (g'_n r_{d-1} + 1)(g'_m r_{d-1} + 1)$.
3. $d > (g'_n r_{d-1} + 1)(g'_m r_{d-1} + 1)$ かつ $(g'_n r_{d-1} + 1)(g'_m r_{d-1} + 1) \mid d$.

のいずれかを満たす. それぞれの場合について以下で考える.

1. $d = (g'_n r_{d-1} + 1)(g'_m r_{d-1} + 1)$ の場合.

条件式を展開して整理すると,

$$\begin{aligned} d &= g'_m g'_n r_{d-1}^2 + g'_m r_{d-1} + g'_n r_{d-1} + 1 \\ d-1 &= g'_m g'_n r_{d-1}^2 + g'_m r_{d-1} + g'_n r_{d-1} \\ g'_m g'_n r_{d-1} r_{mn} &= g'_m g'_n r_{d-1}^2 + g'_m r_{d-1} + g'_n r_{d-1} \\ g'_m g'_n r_{mn} &= g'_m g'_n r_{d-1} + g'_m + g'_n \\ g'_m g'_n (r_{mn} - r_{d-1}) &= g'_m + g'_n \\ r_{mn} - r_{d-1} &= \frac{g'_m + g'_n}{g'_m g'_n}. \end{aligned}$$

左辺は整数であり, 右辺は0より大きな値であるから, この場合 $g'_m g'_n \mid g'_m + g'_n$ を満たさなければならない. これが成り立つのは $g'_m = g'_n = 1$ のときのみであり, $g_m = g_n = r_{mn}$ が従う. これらを上式に代入することで

$$r_{d-1} = r_{mn} - 2, \quad (5.15)$$

が得られる. さらに, $d = (g'_n r_{d-1} + 1)(g'_m r_{d-1} + 1)$ であるから (5.2.1) より,

$$\begin{aligned} d-1 + \gcd(d-1, g_m g_n) &= g_m g_n \\ g_m g'_n (r_{d-1} + \gcd(r_{mn}, r_{d-1})) &= g_m g'_n r_{mn} \\ r_{d-1} + \gcd(r_{mn}, r_{d-1}) &= r_{mn}. \end{aligned}$$

(5.15) より, $\gcd(r_{mn}, r_{d-1}) = 2$ となり $d-1$ は偶数である. また, $\gcd(r_{mn}, r_{d-1}) = 2$, $g_m = g_n$ より $d-1 + 2g_m = g_m^2$ が得られ, この方程式を満たす g_m の値は $1 \pm \sqrt{d}$ である. $g_m = g_n > 0$ であるから, $g_m = g_n = \sqrt{d} + 1$ となる. したがって, この条件を満足する d の値は奇数の2乗で表される.

2. $d > (g'_n r_{d-1} + 1)(g'_m r_{d-1} + 1)$ かつ $d \mid (g'_n r_{d-1} + 1)(g'_m r_{d-1} + 1)$ の場合

条件より, 1 より大きな整数 x に対し $(g'_n r_{d-1} + 1)(g'_m r_{d-1} + 1)/d = x (> 1)$ とおける.
このとき (5.2.1) 式は

$$x g_m g_n = g_m g'_n \gcd(r_{mn}, r_{d-1}) + d - 1,$$

となり, 両辺を $g_m g'_n$ で割ることで

$$x r_{mn} = \gcd(r_{mn}, r_{d-1}) + r_{d-1} \quad (1''),$$

と表せる. ここで, 次の二つの場合を考える.

(a) $r_{mn} = 1$ の場合.

この場合, $g'_m = g_m, g'_n = g_n, d - 1 = g_m g_n r_{d-1}$ かつ (2) 式より $x = r_{d-1} + 1$ であるから, (5.2.1) 式は

$$\begin{aligned} (g_m r_{d-1} + 1)(g_n r_{d-1} + 1) &= d(r_{d-1} + 1) \\ g_m g_n r_{d-1}^2 + g_m r_{d-1} + g_n r_{d-1} + 1 &= d r_{d-1} + d \\ g_m g_n r_{d-1}^2 + g_m r_{d-1} + g_n r_{d-1} &= d r_{d-1} + d - 1 \\ g_m g_n r_{d-1} + g_m + g_n &= d + g_m g_n \\ d - 1 + g_m + g_n &= d + g_m g_n \\ g_m + g_n &= g_m g_n + 1, \end{aligned}$$

となり, この等式が成り立つのは $g_m = 1$ または $g_n = 1$ の場合のみである.

(b) $r_{mn} > 1$ の場合.

この場合, $x > 1$ より, $r_{mn} \leq r_{d-1}$ を満たす. (2) 式の両辺を $\gcd(r_{mn}, r_{d-1})$ で割ると,

$$\begin{aligned} x \frac{r_{mn}}{\gcd(r_{mn}, r_{d-1})} &= 1 + \frac{r_{d-1}}{\gcd(r_{mn}, r_{d-1})} \\ (g'_m r_{d-1} + 1)(g'_n r_{d-1} + 1) \frac{r_{mn}}{\gcd(r_{mn}, r_{d-1})} &= d \left(\frac{r_{d-1}}{\gcd(r_{mn}, r_{d-1})} + 1 \right) \\ \frac{g'_m g'_n r_{d-1}^2 r_{mn} + g'_m r_{d-1} r_{mn} + g'_n r_{d-1} r_{mn} + r_{mn}}{\gcd(r_{mn}, r_{d-1})} &= d \frac{r_{d-1}}{\gcd(r_{mn}, r_{d-1})} + d \\ \frac{(d-1)r_{d-1}}{\gcd(r_{mn}, r_{d-1})} + \frac{g'_m r_{d-1} r_{mn} + g'_n r_{d-1} r_{mn} + r_{mn}}{\gcd(r_{mn}, r_{d-1})} &= \frac{(d-1)r_{d-1}}{\gcd(r_{mn}, r_{d-1})} + \frac{r_{d-1}}{\gcd(r_{mn}, r_{d-1})} + d \\ (g_m + g_n) \frac{r_{d-1}}{\gcd(r_{mn}, r_{d-1})} + \frac{r_{mn}}{\gcd(r_{mn}, r_{d-1})} &= d + \frac{r_{d-1}}{\gcd(r_{mn}, r_{d-1})}, \end{aligned}$$

が得られる. $\gcd(r_{mn}, r_{d-1}) = 1$ であると仮定すると, 上式は $(g_m + g_n - 1)r_{d-1} + r_{mn} = d$ となる. しかしながら $r_{d-1} \mid d - 1$ かつ, この仮定の上では $r_{d-1} > r_{mn}$ で

あるから，式が成り立つには $(g_m + g_n - 1)r_{d-1} = d - 1$ かつ $r_{mn} = 1$ を満たさなければならないが，これは $r_{mn} > 1$ の仮定に反する．よって， $\gcd(r_{mn}, r_{d-1}) > 1$ である．このとき $g'_m + g'_n \leq g'_m g'_n + 1$ は恒等的に成り立つので，上式の左辺第一項は

$$\begin{aligned} (g_m + g_n) \frac{r_{d-1}}{\gcd(r_{mn}, r_{d-1})} &= (g'_m + g'_n) \frac{r_{mn} r_{d-1}}{\gcd(r_{mn}, r_{d-1})} \leq \frac{(g'_m g'_n + 1) r_{mn} r_{d-1}}{\gcd(r_{mn}, r_{d-1})} \\ &= \frac{(g'_m g'_n) r_{mn} r_{d-1}}{\gcd(r_{mn}, r_{d-1})} + \frac{r_{mn} r_{d-1}}{\gcd(r_{mn}, r_{d-1})} \\ &= \frac{d-1}{\gcd(r_{mn}, r_{d-1})} + \frac{r_{mn} r_{d-1}}{\gcd(r_{mn}, r_{d-1})} \\ &\leq \frac{d-1}{2} + \frac{r_{mn} r_{d-1}}{2} \\ &= d-1, \end{aligned}$$

を満たす．さらに， $r_{mn} \leq r_{d-1}$ であるから

$$(g_m + g_n) \frac{r_{d-1}}{\gcd(r_{mn}, r_{d-1})} + \frac{r_{mn}}{\gcd(r_{mn}, r_{d-1})} < d + \frac{r_{d-1}}{\gcd(r_{mn}, r_{d-1})},$$

となり，条件を満たす値は存在しない．

3. $d > (g'_n r_{d-1} + 1)(g'_m r_{d-1} + 1)$ かつ $(g'_n r_{d-1} + 1)(g'_m r_{d-1} + 1) \mid d$ の場合．

条件より，1 より大きな整数 y に対し $d / ((g'_n r_{d-1} + 1)(g'_m r_{d-1} + 1)) = y (> 1)$ とおける．このとき，(5.2.1) 式は，

$$g_m g_n = y(d - 1 + g_m g'_n \gcd(r_{mn}, r_{d-1})),$$

と表せる．しかしながら， $d \perp d-1$ であり y は d の因子 かつ g_m, g_n は $d-1$ の因子であるため， y が $g_m g_n$ の因子となることはなく等式は成り立たない．よって，この場合を満たす値は存在しない．

□

上の補題より， $G_B(d, mn) \mid G_B(d, m) \otimes G_B(d, n)$ であるための必要条件が $\gcd(d-1, m) = 1$ または $\gcd(d-1, n) = 1$ を満たすことであることがわかった．

5.2.2 一般化 de Bruijn ダイグラフの積グラフの分解

ループ付き完全対称ダイグラフ K_d^+ は $G_B(d, d)$ と同型である．本小節では K_d^+ と一般化 de Bruijn ダイグラフのクロネッカー積により得られるダイグラフの同型因子分解を示す．

K_d^+ について， $\gcd(d-1, d) = 1$ であるから，補題 5.10 の条件を満たしている．よって，次のような定理が考えられる．

定理 5.11 $G_B(d, dn) \mid K_d^+ \otimes G_B(d, n)$.

上記の定理は以下の定理の特殊な場合である.

定理 5.12 d, m に対し $\gcd(d-1, m) = 1$ が成り立つと仮定する. このとき,

$$G_B(d, mn) \mid K_m^+ \otimes G_B(d, n).$$

証明 $g_n = \gcd(d-1, n)$, $n' = n/g_n$ とおく. $V(G_B(d, mn))$ の自身への写像 h_β を

$$h_\beta(x) = x + \beta n' \quad (\beta \in \mathbb{Z}_m)$$

と定め, これらにより得られるダイグラフを $H_\beta = h_\beta(G_B(d, mn))$ とおく. さらに, 定理 5.9 の写像 σ を $V(H_\beta)$ から $V(K_m^+ \otimes G_B(d, n))$ への写像として用いることで $x \in V(G_B(d, mn))$ に対し

$$\begin{aligned} \sigma h_\beta(x) &= \sigma((x + \beta n') \bmod mn) \\ &= \left(\left\lfloor \frac{(x + \beta n') \bmod mn}{n} \right\rfloor, (x + \beta n') \bmod mn \bmod n \right) \\ &= \left(\left(\left\lfloor \frac{x + \beta n'}{n} \right\rfloor \right) \bmod m, (x + \beta n') \bmod n \right), \end{aligned}$$

が得られる. まず, 各 H_β での隣接関係が写像 σ で写された $K_m^+ \otimes G_B(d, n)$ 内でも保存されていることを示す. H_β 上で, $h_\beta(x) = x + \beta n'$ と隣接する頂点は $(d(x + \beta n') + k) \bmod mn$ で表される. これらの頂点の σ による写像は

$$\begin{aligned} \sigma((d(x + \beta n') + k) \bmod mn) &= \left(\left\lfloor \frac{(d(x + \beta n') + k) \bmod mn}{n} \right\rfloor, (d(x + \beta n') + k) \bmod mn \bmod n \right) \\ &= \left(\left(\left\lfloor \frac{d(x + \beta n') + k}{n} \right\rfloor \right) \bmod m, (d(x + \beta n') + k) \bmod n \right), \end{aligned}$$

である. $K_m^+ \otimes G_B(d, n)$ において $\left(\left(\left\lfloor \frac{x + \beta n'}{n} \right\rfloor \right) \bmod m, (x + \beta n') \bmod n \right)$ が隣接する頂点は第一項が \mathbb{Z}_m の任意の要素であり, 第二項は $(d(x + \beta n') + k) \bmod n$ を満たすような頂点であるので, 上述で求めた頂点による隣接関係は $K_m^+ \otimes G_B(d, n)$ の隣接関係に含まれている.

次に各 H_β が $K_m^+ \otimes G_B(d, n)$ の弧集合を分割していることを示す. $G_B(d, mn)$ のある頂点 x について, x の弧 $(x, dx + k)$ と h_β で写像したある頂点 $h_\beta(y)$ が接続する弧が等しいと仮定する. このとき接続元の弧は等しくなるため $x = h_\beta(y)$ であり, $x \equiv y + \beta n' \pmod{mn}$ が得られる. また, これらの隣接先が等しいとき, ある $i, j \in \mathbb{Z}_d$ に対して,

$$\begin{aligned} dx + i &\equiv (d(y + \beta n') + j) \pmod{mn} \\ d(x - (y + \beta n')) + i &\equiv j \pmod{mn} \\ i &\equiv j \pmod{mn}, \end{aligned}$$

を得る . y の h_β による隣接先は $d(y + \beta n') = j \pmod{mn}$, $j \in \mathbb{Z}_d$ と表せる . この頂点と y の隣接先の h_β による像が等しいとすると ,

$$\begin{aligned} dy + i + \beta n' &\equiv d(y + \beta n') + j \pmod{mn} \\ (d-1)\beta n' &\equiv i - j \equiv 0 \pmod{mn}. \end{aligned}$$

$d-1 = rg_n$ とおくと $(d-1)n' = rg_n n' = rn$ であるので , $\gcd((d-1)n', mn) = n \cdot \gcd(r, m)$ である . 仮定より , $\gcd(d-1, m) = 1$ であるから $\gcd(r, m) = 1$ であり , 上記の合同式に解があるためには $\beta \equiv 0 \pmod{m}$ を満たす必要がある . これは $\beta \in \mathbb{Z}_m$ であるため $\beta = 0$ を意味し , $x = h_0(y) = y$ を導く . したがって任意の β による弧は互いに素であり , h_β は $K_m^+ \otimes G_B(d, n)$ の弧集合を m 個に分割する . \square

第6章 グラフ上での情報散布問題

本章ではネットワークをモデル化したグラフにおける情報散布について述べる．情報散布の基本的なモデルについて紹介し，本研究で用いる情報散布に関する用語について説明する．

6.1 ネットワーク上での情報散布問題

今日ではより高速な情報処理を行うための技術が必要とされ，そのような要求を満たす方法の一つとしていくつかのプロセッサを相互に繋いだ並列ネットワークのシステム構築がなされている．並列ネットワークにおいてネットワーク上の計算機やプロセッサなどの計算ノードは互いに情報のやり取りをしながら処理を行う．ネットワーク全体の処理能力は各プロセッサの処理能力とネットワークの通信効率に依存し，これら二つの関係は独立ではないため一方の能力が低いと全体の処理能力の低下を招くことになる．また，並列ネットワーク上の各プロセッサが共通のメモリに接続されている場合は各プロセッサとメモリ間での通信を扱うだけでよいが，メモリが分散されている場合にはトポロジ上でデータのやり取りを行う必要がある．そのため，並列ネットワークの処理の高速化を図る方法の一つとしてネットワーク上の各プロセッサ間での情報伝達を効率よく行うことが挙げられ，様々な研究が行われている．

情報散布とはノードが持つ情報をネットワーク上の他のノードへ送信する操作をいい，効率よく情報散布を行うことはネットワーク上の情報伝達を円滑に行うだけでなく，ネットワークの資源を節約できるという観点からも重要である．

代表的な情報散布問題を以下に挙げる．

1. Broadcasting : ネットワーク上のある一つのノードが持つ一つの情報を他のすべてのノードへと送信する．
2. Gossiping : ネットワーク上の各ノードはそれぞれ異なる情報を持ち，すべてのノードがすべての情報を得るように送受信を行う．
3. Scattering : ネットワーク上のある一つのノードが他のすべてのノードに対してそれぞれ異なる情報を送信する．
4. Gathering : ネットワーク上のある一つのノードが他のすべてのノードからそれぞれ異なる情報を受信する．

他にも様々な情報散布問題が [33] や [24] で紹介されている．実際にネットワーク上で情報散布を行う場合，帯域幅やリンクの性質，プロセッサ間で情報を送受信する際の制限などが実行結果に大きく影響を与える．情報散布問題に対する細かな定義および本研究で用いる情報散布問題の設定については次節で示す．

ネットワークのトポロジを選択する際に大切な要素はネットワークのコストと性能という二つの面に大きく分けられる．ネットワークのコストの面では効率的なレイアウトや単純なルーティングアルゴリズムの他に，リンクに用いる線（グラフ上の次数）の最小化および次数が一定であることなど扱いやすい構造を持つことが望ましいとされている．一方で，ネットワークの性能面では小さな直径，一様性や拡張性の高さ，強い連結性などが求められている．しかしながら，例えば直径と最大次数などはトレードオフの関係にあるため，相互結合網のクラスとして様々なグラフが提案されている．

情報散布問題を定式化して考えるために，ネットワークをグラフおよびダイグラフによってモデル化することがなされている．その際，各ノードを頂点，通信リンクを辺あるいは弧に対応させて考える．情報散布の効率を考える上で，まずどのようなトポロジを持つネットワークを用いるかが重要であり，更に用いるネットワーク上で各情報散布問題に対する効率的なアルゴリズムを与える必要がある．特にブロードキャストとゴシップングに対しては広く研究が行われており，様々なグラフクラスでのアルゴリズムが得られている．例として完全グラフ [10]，グリッド [21, 62]，木 [53]，ハイパーキューブ [14, 15, 38] などのグラフクラスが挙げられ，ブロードキャストおよびゴシップングの結果をまとめたものに [3, 16, 24, 31] がある．

de Bruijn 族のグラフへの情報散布アルゴリズムとして，de Bruijn ダイグラフ，Kautz ダイグラフに対するブロードキャストアルゴリズムが提案されており，de Bruijn ダイグラフに対してはマルチソースブロードキャストのアルゴリズムも提案されている [37]．本研究では de Bruijn ダイグラフに対して異なるマルチソースブロードキャストのアルゴリズムを提案し，Kautz ダイグラフに対しても cycle-rooted tree を用いたいくつかのアルゴリズムを提案する．

6.2 情報散布問題の設定

この節では情報散布問題に関する用語を定義し，本研究において用いるネットワークモデルについて述べる．

同期システム 本研究では同期システムを用いる．ネットワーク全体に対し大域時計が設定されており，各ノードは同期を取りながら情報の送受信を行なう．ネットワーク内での情報の送受信は store and forward 型を考える．情報の大きさは固定であるとし，一つの情報の送受信を行なうための単位時間をラウンドとして表す．すなわち，情報散布の終了までにかか

るラウンドが少ないほど，効率的な情報散布であると考えられる．

ネットワークをグラフまたはダイグラフにより定式化するためにプロセッサを頂点，通信リンクを辺または弧によりモデル化する．ある二つのプロセッサ p_1, p_2 間の通信リンクに対し，1 ラウンドに p_1 から p_2 への通信または p_2 から p_1 への通信のどちらかのみを許可するものは半 2 重通信 (half-duplex) と呼ばれ，両方向からの通信を同時に許可するものは全 2 重通信 (full-duplex) と呼ばれる．前者は主にグラフに，後者はダイグラフに対するモデル化に用いられる．本研究ではダイグラフでネットワークをモデル化して考えるため，通信リンクとして常に一方向のみからの情報伝達が可能な半 2 重通信を考える．

情報散布において，各頂点が 1 ラウンドで使用可能なリンクの総数を制限する必要があることは自然な考えである．特に，1 ラウンドで各頂点が高々一つの隣接辺のみを扱えるモデルを processor-bound または whispering, すべての隣接辺を扱えるものを link-bound または shouting という．本研究では以下で定義されるモデルを用いる．

同時送受信モデル (simultaneous send/receive model) [1] 同時送受信モデルとは，以下の制約条件を同時に満たす送受信モデルである．

- 各頂点は 1 ラウンドに高々一つの情報を受信できる．
- 各頂点は 1 ラウンドに高々一つの情報を送信できる．
- 各頂点は 1 ラウンドに送信と受信の両方を同時に行なうことができる．

また，各ノードは各情報が識別可能であり，自分の持っている情報が複数ある場合にはどの情報を送信するかを選択できるものとする．さらに，今回のモデルではネットワーク全体を管理する外部サーバーが存在しているものとし，これにより各頂点は外部からの指示によっても情報を送信することが可能であるとする．

マルチソースブロードキャストイング 本論文では情報散布問題の一つであるマルチソースブロードキャストイングについて研究を行う．ブロードキャストイングはネットワーク上のある一つのノードが持つ情報を他のすべてのノードへ散布する問題であり，ゴシップングはネットワーク上のすべてのノードがそれぞれ持つ異なる情報をすべての頂点への散布する問題である．これら二つの問題は，ネットワーク上の情報をすべての頂点へと送信する問題としては extremal case に属するものである．そのため，いくつかのノードが持っている情報をすべてのノードに散布することがより実際的な問題であると考えられる．マルチソースブロードキャストイングとはネットワーク上にある複数個のノードが持つ情報を，すべての頂点がすべての情報を持つように情報の送受信を行なうという問題であり，[41] で提案された．マルチソースブロードキャストイングをダイグラフでモデル化した場合，次の定義のように定式化することができる．

定義 6.1 D を頂点数 n のダイグラフとし, $X = \{x_1, x_2, \dots, x_k\} \subset V(D)$, $1 \leq k \leq n$ とする. X の各頂点をソース頂点, あるいは単にソースと呼び, 頂点 x_i は情報 m_i を持つ. ここで $i \neq j$ に対し $m_i \neq m_j$ である. $V(D)$ のすべての頂点がすべての情報 m_1, m_2, \dots, m_k を持つように情報の送信を行なう問題をマルチソースブロードキャストイングと呼ぶ.

ブロードキャストイングはグラフ上のある一つの頂点から他のすべての頂点への情報散布 (one to all) であり, ゴシッピングはグラフ上のすべての頂点からすべての頂点への情報散布 (all to all) である. これら二つの問題は, ネットワーク上の情報をすべての頂点へと送信する問題としては extreme case に属するものである. 一方で, マルチソースブロードキャストイングはグラフ上のいくつかの頂点からすべての頂点への情報散布 (some to all) であり, この問題はブロードキャストイングとゴシッピングを含む情報散布問題の体系を構成すると考えられる.

ブロードキャストイングは一つの情報を扱う情報散布であり, 複数個の情報を扱う場合に対しては, 各頂点が1ラウンド中に二つ以上の情報を同時に受信する場合について考えておらず, そのような場合が起きないために単一のブロードキャストイングを複数回行なう必要がある. それに対し, マルチソースブロードキャストイングは複数のブロードキャストイングを独立・並行に行なっていると考えることもできる.

情報散布を行なう際に, 各頂点は新しい情報を受信したらすぐにその情報を送信できる状態になっていることが望ましい. しかしながら, マルチソースブロードキャストイングのようにネットワーク上で複数の情報を同時に扱う場合, 受信済みの情報を送信しきれずに新たな情報を受信してしまうことが考えられる. これらのことに対して, 次の二つの用語を定義する.

インターバル v をダイグラフ D の頂点とする. v が情報 m_i, m_j をこの順番で受信し, かつ間に他の情報の受信はないものとする. このとき, m_i を受信してから m_j を受信するまでのラウンドの差を v における m_i と m_j のインターバルと呼び, $I_v(m_i, m_j)$ で表す.

遅延 ダイグラフ D の頂点 v に対し, v が情報 m_i を受信したラウンドを r_i , v が子へ m_i の送信を始めたラウンドを s_i とおく. このとき, $(s_i - 1) - r_i$ で表されるラウンドの差を v における m_i の遅延と呼ぶ.

インターバル, 遅延の定義により, 次の定理は明らかである.

定理 6.2 ダイグラフ D の頂点 v と v が受信する情報 m_{k-1}, m_k に対し, $I_v(m_{k-1}, m_k) = c$, $\text{od } v = d$ とおく. さらに, v において m_{k-1} の遅延は0とする. このとき, m_k の遅延が0となるための必要十分条件は $c \geq d$ を満たすことである.

マルチソースブロードキャストイングを行なう際に, 送信する情報を k (≥ 2) としたとき, 情報の集合を $M = \{m_1, m_2, \dots, m_k\}$ と表す.

第7章 de Bruijn 族グラフ上でのマルチソースブロードキャストティング

本章では de Bruijn ダイグラフおよび Kautz ダイグラフに対し cycle-rooted tree を用いたマルチソースブロードキャストティングのアルゴリズムを与え、その性能を評価する。

複雑なグラフ上でのマルチソースブロードキャストティングを行なう場合に、そのグラフに対する既存のブロードキャストティングや、含まれている部分グラフに対する (マルチソース) ブロードキャストティングの手法を利用する場合がある。本論文では cycle-rooted tree 上でのマルチソースブロードキャストティングについて扱うが、cycle-rooted tree の構造から木の上でのブロードキャストにかかる時間を与えることが重要である。本章で扱う二つの根付き木に対して必要なブロードキャストティング時間は以下で与えられる。

定理 7.1 (Hromkovic, Klasing Monien and Peine [33], Lemma 5.2.4) $CT(d, h)$ のブロードキャストティング時間は dh ラウンドである。

定理 7.2 (入野, 田中, 河合, 大澤, 柴田 [37], 系 1) $CT^-(d, h)$ のブロードキャストティング時間は $dh - 1$ ラウンドである。

7.1 de Bruijn ダイグラフ上でのマルチソースブロードキャストティング

de Bruijn ダイグラフに対して、全域木を用いたブロードキャストティングアルゴリズムが Heydemann らにより提案されている [32]。このアルゴリズムによって得られるブロードキャストティング時間の上界は次で表される。

定理 7.3 (Heydemann, Opatrny and Sotteau [32], Theorem 3.5) 次のラウンド数を満たす de Bruijn ダイグラフ上のブロードキャストティングアルゴリズムが存在する。ただし、 $b(B(d, n))$ とは $B(d, n)$ の任意の頂点からのブロードキャストにかかるラウンド数の中で最大のものである。

$$b(B(d, n)) \leq n \left(\frac{5}{4} \lceil \log_2 d \rceil + 3 \right)$$

本論文では得られたマルチソースブロードキャストティングアルゴリズムの性能の比較対象として、[32] のアルゴリズムを情報の総数分繰り替えした結果を用いる。

7.1.1 全域 cycle-rooted tree を用いたアルゴリズム

入野らは定理 4.1 で得られる d 個の完全 d 進 loop-rooted tree をもとに一つの全域 cycle-rooted tree を構成し, その上でマルチソースブロードキャストを行う手法を提案した [37]. この方法で構成される全域 cycle-rooted tree の root-cycle は d 個の loop-rooted tree を有向パスで繋げた形となっている. また, cycle-vertex から cycle-vertex への有向パス上に現れる collateral tree は同型であり, ループ頂点から離れるほど collateral tree の深さは小さくなる. これらの性質を利用し, 情報をループ頂点を持つように一定の間隔で配置することで遅延が発生せずに情報散布を行うアルゴリズムを与えている. 上記の手法で得られたマルチソースブロードキャストにかかる時間の上限は次のように与えられる.

定理 7.4 (入野, 田中, 河合, 大澤, 柴田 [37], 定理 3) de Bruijn ダイグラフ $B(d, n)$ 上で高々 d 個の情報をマルチソースブロードキャストするために必要なラウンド数は $n < d$ のとき高々 $d^2 + 2dn - 2d - 1$ であり, $n \geq d$ のとき高々 $2dn - d$ である.

7.1.2 cycle-rooted tree による分解を用いたアルゴリズム

この小節では $B(d, n)$ 上での loop-rooted tree を用いたマルチソースブロードキャストのアルゴリズムを提案する. 定理 4.1 より, $B(d, n)$ は d 個のループ頂点 (x, x, \dots, x) , $0 \leq x \leq d-1$ を root-cycle とするような深さ n の loop-rooted tree によって同型因子分解される. これらの loop-rooted tree をループ頂点 (x, x, \dots, x) の文字に応じて $SPLRT_i$ と定義する.

各頂点は 1 ラウンドに高々一つの情報のみを受信できるため, 複数の情報を同時に送受信をする場合には同ラウンド中に二つ以上の情報が一つの頂点に送信されないように配慮する必要がある. これに関して, 各 $SPLRT_i$ でそれぞれの頂点に追加のラベルを与えることで頂点が 1 ラウンド中に二つ以上の情報を受信することがないようにアルゴリズムを以下で構成する.

まず, $B(d, n)$ の頂点集合を次のように分割する.

$$V(x_1, x_2, \dots, x_{n-1}) = \{(x_1, \dots, x_{n-1}, \alpha) \mid 0 \leq \alpha \leq d-1\}, 0 \leq x_i \leq d-1, 0 \leq i \leq n-1.$$

このとき, 頂点 $(\beta, x_1, x_2, \dots, x_{d-1})$, $0 \leq \beta \leq d-1$ は $(x_1, x_2, \dots, x_{d-1}, \alpha)$ で表されるすべての頂点に隣接することから, 次は明らかである.

補題 7.5 $B(d, n)$ に対し, $V(x_1, x_2, \dots, x_{d-1})$ に属する d 個の頂点はすべて同じ頂点から隣接されている.

次に, $V(x_1, x_2, \dots, x_{d-1})$ から整数集合 $\mathbb{Z}_d = \{1, 2, \dots, d\}$ への bijection について考えると次が成り立つ.

補題 7.6 任意の $V(x_1, x_2, \dots, x_{d-1})$ に対し, 次の条件を満たすような \mathbb{Z}_d への d 個の bijection L_0, L_1, \dots, L_{d-1} が存在する .

$$i \neq j, 0 \leq i, j \leq d-1 \Rightarrow L_i(v) \neq L_j(v), \forall v .$$

例としてある bijection L に対し, $L^i(v) = (L(v) + i) \bmod d$ を定義すると L, L^1, \dots, L^{d-1} は補題 7.6 の条件を満たす d 個の bijection である .

ここで, 各 $SPLRT_i$ において, それぞれの頂点に bijection L_i によって写像される値と等しいラベルを与えることを考える . すると, 各頂点は $SPLRT_i$ 毎に異なる値のラベルを持つ . また $SPLRT_i$ 内において, 葉でないすべての頂点は隣接先の d 個の頂点が 1 から d までの互いに異なるラベルを持つことがいえる . $SPLRT_i$ は完全 d 進 loop-rooted tree であるから, その上でのブロードキャストについて, 次が成り立つ .

定理 7.7 $SPLRT_i$ の cycle-vertex (i, i, \dots, i) が持つ情報をブロードキャストするために必要なラウンド数は $dn - 1$ である .

このとき, $SPLRT_i$ は完全 d 進で葉の深さはすべて n であるから, 葉以外の頂点は情報を受信した次のラウンドからすぐに子への送信を始めさえすれば, どのような順番で情報の送信を行なったとしても上記のラウンド数ですべての頂点が情報を受信することができる .

今, 各 $SPLRT_i$ 上で並列的に情報を一つずつブロードキャストすることを考える . もし d 個の $SPLRT_i$ 上で, 対応する cycle-vertex (i, i, \dots, i) からのブロードキャストを各頂点が同ラウンドに二つ以上情報を受信することがないように行なうことができれば, $SPLRT_i$ 上でブロードキャストを行なう度に $B(d, n)$ 上で高々 d 個までの情報をすべての頂点が得られると考えられる . そのためには, まず d 個のループ頂点 (i, i, \dots, i) にそれぞれ d 個の情報を送信する必要がある . そのために必要なラウンド数は次で与えられる .

定理 7.8 (Irinu, Tanaka, Kawai, Osawa and Shibata [37], 系 1) $B(d, n)$ 上に与えられた d 個の情報を m_0, m_1, \dots, m_{d-1} とおく . このとき, 各 m_i を頂点 (i, i, \dots, i) に送信するために必要なラウンド数は高々 n ラウンドである .

定理 7.8 により各ループ頂点に情報が送信された後で, 各 $SPLRT_i$ 上でのブロードキャストを考える . 今, 補題 7.6 の条件を満たし, かつすべての $0 \leq i \leq d-1$ に対して $L_i(i, i, \dots, i) = d$ を満たす bijection L_0, L_1, \dots, L_{d-1} について考える . $B(3, 3)$ におけるそのような bijection の例を図 7.1 に示す . このような bijection に対し, 次が成り立つ .

補題 7.9 各 $SPLRT_i$ 上でのブロードキャストに対して頂点 v が情報 m_i を受け取るラウンドを $r_i(v)$ とおく . このとき, bijection L_i が各ループ頂点 (i, i, \dots, i) に対して $L_i((i, i, \dots, i)) = d$ を満たすならば, 総ラウンド数が $dn - 1$ であり, 任意の頂点 v に対して $L_i(v) = (r_i(v) \bmod d)$ であるような $SPLRT_i$ 上のブロードキャストアルゴリズムが存在する .

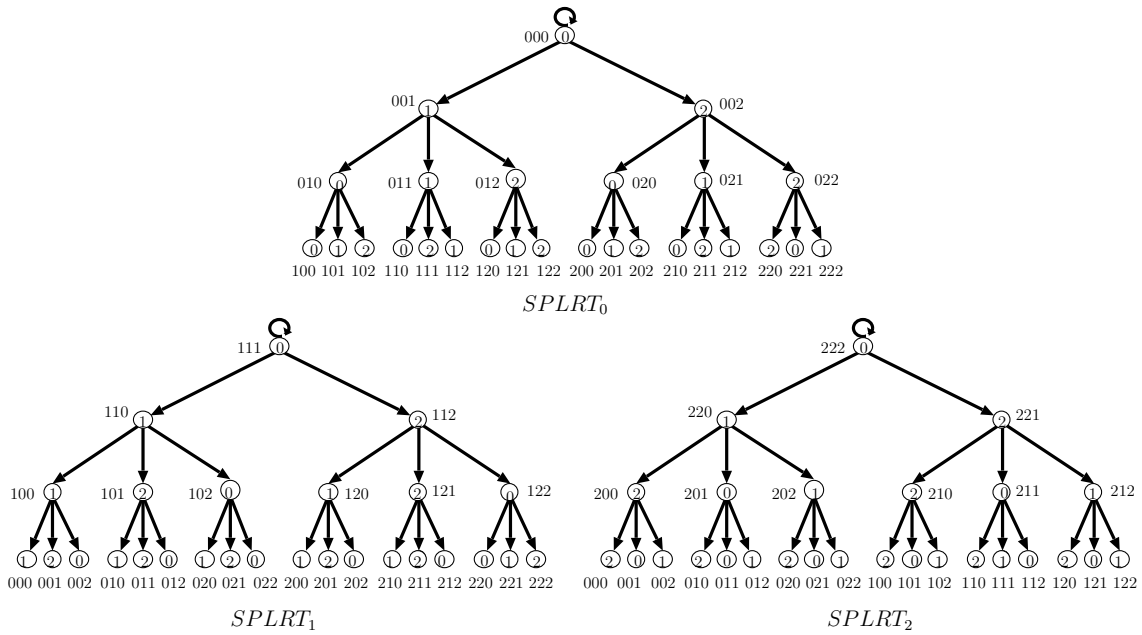


図 7.1: $B(3, 3)$ 上での bijection の例

証明 定理 7.7 より, $SPLRT_i$ 上の葉以外の頂点 u に対し $r_i(u)$ ラウンドから $r_i(u) + d - 1$ ラウンドの間に d 個の子すべてに上記の条件を満たすように情報の送信ができればよい. $u = (x_1, x_2, \dots, x_n)$ とおくと, u の子は $(x_2, \dots, x_n, \alpha)$, $0 \leq \alpha \leq d - 1$ で表されるため, 同じ頂点部分集合 $V(x_2, \dots, x_n)$ に含まれる. よって補題 7.6 により得られる bijection L_i によって写像される値はすべて異なる値である. したがって, u の子には $L_i(w) = (r_i(u) \bmod d)$ を満たす w がただ一つ存在する. 同様に u の子には $L_i(v) = (r_i(u) + j \bmod d)$, $0 \leq j \leq d - 1$ を満たすような v が一つずつ存在するため, 情報を受信してから d ラウンドで上記の条件を満たすようにすべての子へ情報を送信することが可能である. さらに仮定により $SPLRT_i$ 上の cycle-vertex (i, i, \dots, i) は $d - 1$ 個の子にそれぞれ 1 から $d - 1$ までの値が写像されるので, $d - 1$ ラウンドで条件を満たすように子への送信が可能である. 以上のことから, 条件を満足するようなアルゴリズムは存在する. \square

各 $SPLRT_i$ に対して, 補題 7.9 から得られるアルゴリズムを同時に行なうことを考える. このとき, 次が成り立つ.

定理 7.10 $B(d, n)$ の二つの異なる $SPLRT_i, SPLRT_j$ ($0 \leq i, j \leq d - 1, i \neq j$) と任意の頂点 v に対し, $r_i(v) \neq r_j(v)$ が成り立つ.

証明 $SPLRT_i$ において, 頂点 v が情報 m_i を受信するラウンド $r_i(v)$ は $r_i(v) = (L_i(v) \bmod d)$ を満たす. このとき, 各 L_i は補題 7.6 により得られる bijection であるから $L_i(v) \neq L_j(v)$ を満たす. さらに $L_i(v), L_j(v) \in \mathbb{Z}_d$ であるから, $L_i(v) \neq (L_j(v) \bmod d)$ が成り立ち $r_i(v) \neq (r_j(v) \bmod d)$ が従う. よって, どの二つの $SPLRT_i$ の間においても v が情報を得るラウン

ドは d を法として合同とならず, これは $r_i(u) \neq r_j(v)$ を導く. \square

定理 7.10 より, 任意の異なる二つの $SPLRT_i$ 上においてどの頂点もそれぞれ異なるラウンドで情報を得ることがいえ, これは与えられた方法で各 $SPLRT_i$ 上で同時にブロードキャストを行う際にどの頂点も同ラウンドに二つ以上の情報を受信することがないということを示している. さらに $SPLRT_i$ は完全 d 進 loop-rooted tree による同型因子分解であるから, 各頂点はいずれか一つの $SPLRT_i$ からのみ弧が出ている. よって各頂点は同時に二つ以上の情報を送信を行なうことはない. 以上より次が成り立つ.

定理 7.11 $B(d, n)$ に対し各ループ頂点の一つずつ異なる情報を持っていると仮定する. このとき, $B(d, n)$ のすべての頂点に d 個すべての情報を送信するために必要なラウンド数は高々 $dn - 1$ である.

さらに定理 7.8, 定理 7.10 より, 高々 d 個の情報をマルチソースブロードキャストするには高々 $(d+1)n - 1$ ラウンド必要であることがわかる. 情報の総数が d よりも大きな場合には, 情報の衝突を避けるため情報をまとめて d 個ずつ送信していくことにする. これらの方法により $B(d, n)$ 上でマルチソースブロードキャストを行うために必要なラウンド数は次のようになる.

定理 7.12 $X = \{x_1, x_2, \dots, x_k\} \subset V(B(d, n))$, $1 \leq k \leq d^n$ を $B(d, n)$ 上のソース頂点とする. このとき $B(d, n)$ 上で k 個の情報をマルチソースブロードキャストするために必要なラウンド数は高々 $\lceil k/d \rceil (d+1)n - 1$ である.

7.2 Kautz ダイグラフ上でのマルチソースブロードキャスト

Kautz ダイグラフに対しても Heydemann らによるブロードキャストアルゴリズムが提案されている [32]. このアルゴリズムによって得られるブロードキャスト時間の上界は次で表される.

定理 7.13 (Heydemann, Opatrny and Sotteau [32], Proposition 2.12) 次のラウンド数を満たす Kautz ダイグラフ上のブロードキャストアルゴリズムが存在する. ただし, $b(K(d, n))$ とは $K(d, n)$ の任意の頂点からのブロードキャストにかかるラウンド数の中で最大のものである.

$$\begin{aligned} b(K(2, n)) &\leq 2n, \\ b(K(3, n)) &\leq 3n, \\ b(K(d, n)) &\leq \begin{cases} \frac{(d+3)(n+1)}{2}, & \text{if } 4 \leq d \leq 12, d \neq 9 \\ \min\{2n \lceil \log_2 d \rceil, 3n \lceil \log_3 d \rceil\}, & \text{otherwise.} \end{cases} \end{aligned}$$

7.2.1 大きな root-cycle を持つ全域 cycle-rooted tree の構成

ここでは定理 4.16 より得られる Kautz ダイグラフの同型因子分解を用いてより大きな root-cycle を持つ全域 cycle-rooted tree を構成するための方法をいくつかの小さ節に分けて述べる .

S の定義と性質

定理 4.16 の方法により $K(d, n)$ を同型因子分解することで得られる各因子 F_i ($1 \leq i \leq d$) に対し, すべての葉を消去したダイグラフをそれぞれ S_i ($1 \leq i \leq d$) と定義し S_i の集合を S と表す . また, 各 S_i に対し dv_i を根とする collateral tree を $Co-dT_i^-$ と表し sv_i を根とする collateral tree を $Co-sT_i^-$ と表す . S_i も F_i と同様に葉以外の出次数はすべて d である . 図 7.2 に S_i の構造を示す . このとき, 次の補題が成り立つ .

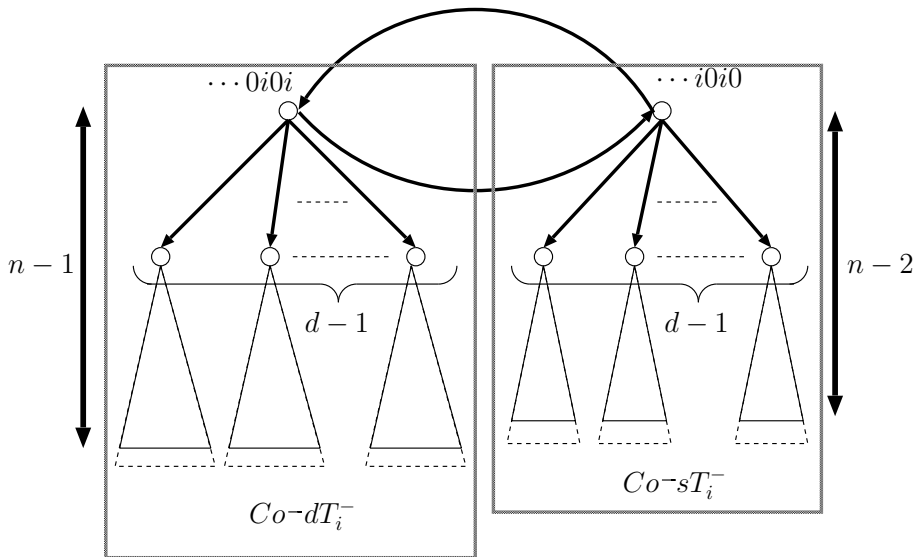


図 7.2: S に属する要素の構造

補題 7.14 S の任意の二つの異なる要素 S_i, S_j ($1 \leq i, j \leq d, i \neq j$) に対し, $V(S_i) \cap V(S_j) = \emptyset$ であり, かつ $\bigcup_{1 \leq i \leq d} V(S_i) = V(K(d, n))$ である .

証明 S_i は頂点集合として dv_i から $K(d, n)$ 内での長さが $n-1$ の任意の有向パスに含まれている頂点のみを持っている . したがって S_i に含まれる頂点は $v_0 = i$ または $v_0 = 0, v_1 = i$ を満たすようなすべての頂点 $v = (v_0, v_1, \dots, v_{n-1})$ である . よって $i \neq j$ のとき $V(S_i) \cap V(S_j) = \emptyset$ が成り立つ .

次に $\bigcup_{1 \leq i \leq d} V(S_i) = V(K(d, n))$ を示す . $K(d, n)$ の頂点の中に $v_0 = i$ であるような頂点は d^{n-1} 個存在し, $v_0 = 0, v_1 = i$ であるような頂点は d^{n-2} 個存在する . 各 S_i の頂点数は

$Co-dT_i^-$ 及び $Co-sT_i^-$ の形から $d^{n-1} + d^{n-2}$ であるので

$$\begin{aligned} \sum_i |S_i| &= d(d^{n-1} + d^{n-2}) \\ &= d^n + d^{n-1} \\ &= |V(K(d, n))|, \end{aligned}$$

が成り立つ．したがって $K(d, n)$ 内で $v_0 = i$ または $v_0 = 0, v_1 = i$ であるようなすべての頂点が S_i に含まれており, $\bigcup_{1 \leq i \leq d} V(S_i) = V(K(d, n))$ が成り立つ． \square

補題 7.14 からダイグラフ $\bigcup_{1 \leq i \leq d} S_i$ は Kautz ダイグラフの全域非連結部分ダイグラフであり, 成分として S_1, S_2, \dots, S_d を持つ．それらの各成分は cycle-rooted tree であるから任意の cycle-vertex から同じ成分内の頂点への有向パスは一意に存在する．これはすなわち, $\bigcup_{1 \leq i \leq d} S_i$ の異なる成分間に弧を加えることで S のすべての要素の cycle-vertex を少なくとも一つずつ含むような有向サイクルが構成可能であれば, それを root-cycle とする全域 cycle-rooted tree が構成できることを意味している．

S の要素間の隣接関係

ここでは S の異なる二つの要素間に存在する $K(d, n)$ 上の弧の性質について述べる． S の異なる二つの要素間に存在する弧の性質について述べ, その後で実際に S の各要素の形を崩さないような全域部分ダイグラフの構成方法を与え, 得られたダイグラフから全域 cycle-rooted tree が得られることを示す． S の各要素は葉以外のすべての頂点の出次数が d であるため, 二つの要素間には一方の葉から出ている弧しか存在しない．そのため S の各要素の cycle-vertex を含む有向サイクルを構成するにはその cycle-vertex へ接続する弧が重要な役割を果たす．

S の各要素の cycle-vertex へ接続する弧に関して, n が奇数であるとき次のような性質が得られる．

補題 7.15 n が奇数のとき S の異なる二つの要素 S_i, S_j ($1 \leq i, j \leq d$) 間で以下が成り立つ．

- (i) $Co-sT_i^-$ の葉から sv_j への弧がただ一つ存在する．
- (ii) $Co-dT_i^-$ の葉から dv_j への弧がただ一つ存在する．

証明 n は奇数であるから, $sv_i = (0, i, \dots, i, 0)$ であり $dv_i = (i, 0, \dots, 0, i)$ である． $Co-sT_i^-$ は $CT^-(d, n-2)$ と同型であるのでその各葉はそれぞれ $(i, 0, k, *, \dots, *)$ ($k \neq 0, i$) と表され, この中に $(i, 0, j, 0, \dots, 0, j)$ で表される頂点が存在する．この頂点は $(0, j, 0, \dots, 0, j, 0)$, すなわち, sv_j への弧を持つ．よって $1 \leq i \leq d, i \neq j$ であるすべての i に対し $Co-sT_i^-$ の

葉から sv_j への弧が存在する。 $K(d, n)$ は d 正則であるから、これら $d-1$ 本の弧と dv_j から sv_j への弧 (dv_j, sv_j) を除くと他に sv_j への弧は存在しない。したがって (i) が成り立つ。

同様に、 $Co-dT_i^-$ は $CT^-(d, n-1)$ と同型であるからその葉の中に頂点 $ij0j \cdots j0$ が存在する。この頂点は $j0 \cdots j0j$ 、すなわち、 dv_j への弧を持つ。 dv_j への弧はこれら $d-1$ 本の弧および (sv_j, dv_j) を除いて存在しない。したがって (ii) が成り立つ。□

図 7.3 に S_i の葉と S_j の根の間の隣接関係を示す。

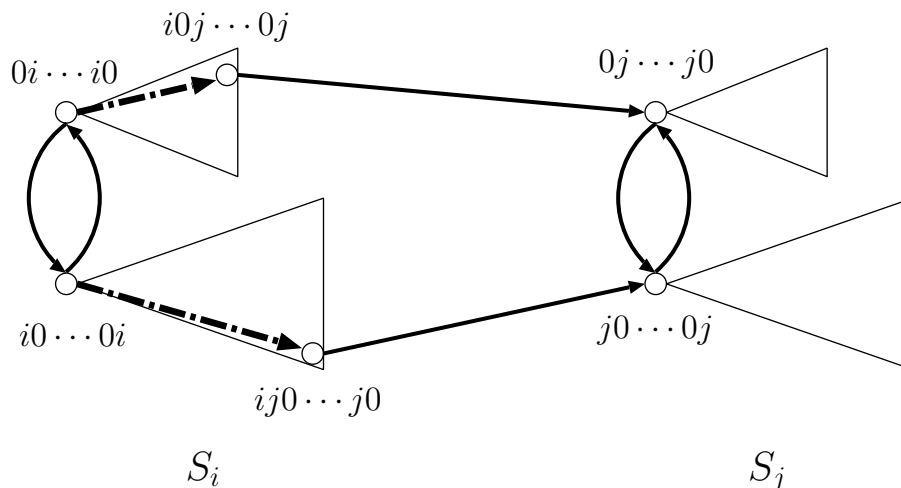


図 7.3: 二つの異なる $S_i, S_j \in \mathcal{S}$ 間の隣接関係 (n が奇数の場合)

n が偶数の場合についても類似した性質を持つ。

補題 7.16 n が偶数のとき、 \mathcal{S} の異なる二つの要素 S_i, S_j ($1 \leq i, j \leq d$) 間で以下が成り立つ。

- (i) $Co-sT_i^-$ の葉から dv_j への弧がただ一つ存在する。
- (ii) $Co-dT_i^-$ の葉から sv_j への弧がただ一つ存在する。

証明 n が偶数であるから、 $sv_i = (i, 0, \dots, i, 0)$ で表される。 sv_i の子は $(0, i, \dots, i, 0, k)$ ($k \neq 0, i$) で表され、 $Co-sT_i^-$ は $CT^-(d, n-2)$ と同型であるので、その各葉は $(i, 0, k, *, \dots, *)$ でそれぞれ表される。この中には $(i, 0, j, 0, \dots, j, 0)$ と表される頂点が存在し、これは $(0, j, 0, \dots, 0, j)$ 、すなわち、 dv_j への弧を持つ。したがって、 $Co-sT_i^-$ の葉から dv_j への弧が存在する。ここで、 $1 \leq i \leq d, i \neq j$ であるすべての i について同様に $Co-sT_i^-$ の葉から dv_j の根への弧が存在する。補題 7.14 より、これら $d-1$ 個の弧はすべて異なるものであり $K(d, n)$ の入次数は d であるから上述の $d-1$ 本の弧および (sv_j, dv_j) 以外に dv_j への弧は存在しない。したがって (i) が成り立つ。

同様に、 dv_i の子は $(i, 0, \dots, 0, i, k)$ で表される。 $Co-dT_i^-$ は $CT^-(d, n-1)$ と同型であるから、その各葉は $(i, k, *, \dots, *)$ で表され、この中には $(i, j, 0, \dots, 0, j)$ で表される頂点が存在する。

在する．この頂点は $(j, 0, \dots, j, 0)$, すなわち, sv_j への弧を持っている．さらに sv_j への弧はこれら $d - 1$ 本の弧と (dv_j, sv_j) 以外には存在しない．したがって (ii) が成り立つ． \square

図 7.4 に S_i の葉と S_j の根の間の隣接関係を示す．

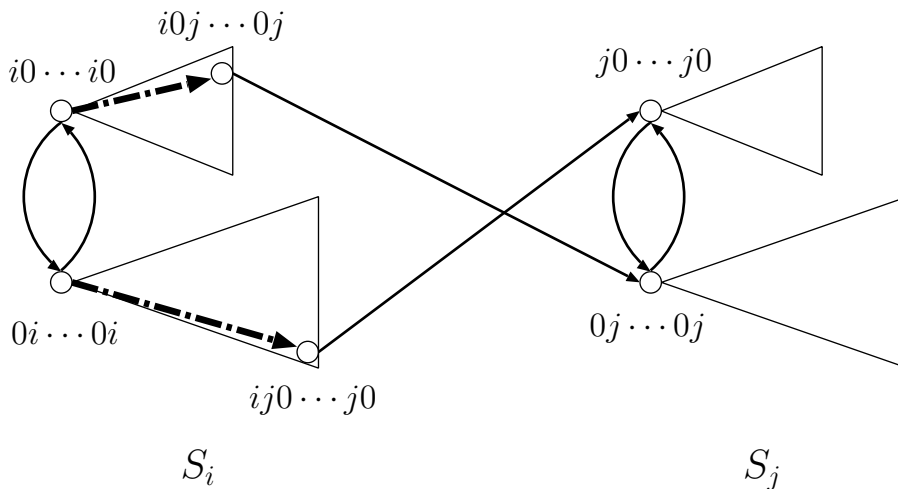


図 7.4: 二つの異なる $S_i, S_j \in S$ 間の隣接関係 (n が偶数の場合)

補題 7.15, 補題 7.16 より, 二つの異なる $S_i, S_j \in S$ ($1 \leq i, j \leq d$) に対しては $K(d, n)$ のパラメータに関わらず S_i の root-cycle の頂点から S_i の葉までの有向パスをたどり, そこから S_j の root-cycle の頂点への弧を繋いだような有向パスが必ず存在することが明らかとなった．

S を用いた全域 cycle-rooted tree の構成

ここでは実際に $K(d, n)$ の全域 cycle-rooted tree を S をもとにして構成するための方法を示す．

S の各要素が持つ弧と, $K(d, n)$ 上で dv_i または sv_i ($i = 1, 2, \dots, d$) へ接続する弧からなる弧集合を $A(S)$ として定義する．また, $A(S)$ の弧誘導部分ダイグラフ $\langle A(S) \rangle$ 上において, 各 S_i の cycle-vertex を少なくとも一つずつ含むような有向サイクルの集合を RC として定義する．このような RC の要素 rc に対し各頂点の入次数が 1 となるように rc に含まれない弧を削除していくことで $\langle A(S) \rangle$ から rc を root-cycle とする全域 cycle-rooted tree が構成できる．

$A(S)$ に含まれる弧の中で, S の要素 S_i に含まれる cycle-vertex でない頂点へ接続する弧はその S_i 内での親から接続する弧しか存在しない．したがって $\langle A(S) \rangle$ から全域 cycle-rooted tree を構成するときこれらの弧はすべて含まれていなければならない．一方, 各 S_i の cycle-vertex に接続する弧は補題 7.15, 補題 7.16 より d 本ずつ存在するため, そのそれぞれに対し $d - 1$ 本の弧を削除する必要がある．ここで, $\langle A(S) \rangle$ の有向サイクル C に対し

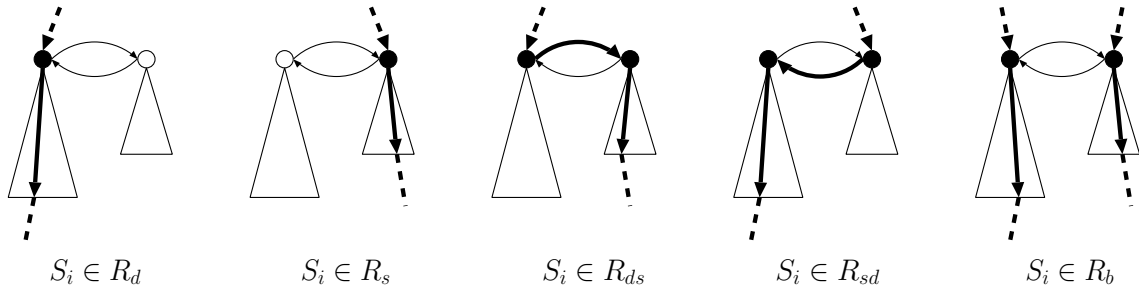


図 7.5: 各分割集合に属する S_i と C の関係

S の各要素 S_i 上の root-cycle を構成する要素が C 内にどのように含まれるかによって S を以下の 6 つの集合に分割することができる .

$$\begin{aligned}
 R_d &= \{S_i \mid dv_i \in V(C) \text{ かつ } sv_i \notin V(C)\}. \\
 R_s &= \{S_i \mid sv_i \in V(C) \text{ かつ } dv_i \notin V(C)\}. \\
 R_{ds} &= \{S_i \mid (dv_i, sv_i) \in A(C)\}. \\
 R_{sd} &= \{S_i \mid (sv_i, dv_i) \in A(C)\}. \\
 R_b &= \{S_i \mid dv_i, sv_i \in V(C) \text{ かつ } (dv_i, sv_i), (sv_i, dv_i) \notin A(C)\}. \\
 R_x &= \{S_i \mid dv_i, sv_i \notin V(C)\}.
 \end{aligned}$$

R_x を除いたそれぞれの分割集合に属する S の要素に対し , root-cycle のどの部分が C に含まれているかを図 7.5 に示す . C に含まれる cycle-vertex は黒点で , C に含まれる cycle-vertex の弧は太線で示されている . RC の要素 rc に対し $|R_x| = 0$ であることは明らかである . また , この分割を行なったとき rc に含まれる cycle-vertex に対しては rc 上で接続される弧以外を削除することで入次数は 1 となる . rc に含まれていない各 S_i の cycle-vertex に対しては , S_i のもう一方の cycle-vertex からの弧のみを残し他の $d - 1$ 個の辺を削除することで入次数 1 となり , rc を root-cycle とする全域 cycle-rooted tree が構成できる .

$\langle A(S) \rangle$ の有向サイクル C に対し , S を上記の集合に分割することで C を順序列 s_1, s_2, \dots, s_x を用いて表現することができる . 各 s_i ($1 \leq i \leq x$) は S の要素であり s_i から s_{i+1} への有向パスはお互いが属している分割集合の条件を満たすものとする . このとき R_b に属する要素は構成される部分ダイグラフ内で二つの cycle-vertex が接続されていないため , 順序列内に 2 回出てくることに注意する必要がある .

上記の 6 つの集合に S を分割したとき , n の偶奇に応じて次の二つの定理が成り立つ .

定理 7.17 n を奇数とする . 有向サイクル C が RC に含まれているための必要十分条件は $|R_x| = 0$ であり , かつ次の条件のいずれかを満たすことである .

- (i) $|R_{ds}| = |R_{sd}| \neq 0$.
- (ii) $|R_d| = d$.
- (iii) $|R_s| = d$.

証明 RC に含まれる有向サイクル C が (i), (ii), (iii) の条件のいずれも満たしていないと仮定する．このとき考えられるのは以下の3通りである．

- $|R_{ds}| > 0, |R_{sd}| > 0$ かつ $|R_{ds}| \neq |R_{sd}|$.
- $|R_{ds}| = |R_{sd}| = 0$ かつ, $|R_d| > 0, |R_s| > 0$.
- $|R_{ds}| = |R_{sd}| = 0$ かつ, $|R_b| > 0$.

どの場合にも矛盾が生じることを示す．

一つ目の場合, $|R_{ds}| > |R_{sd}|$ と仮定する． C を順序列で表すと, n は奇数であるから補題 7.15 より C 内で二つの R_{ds} の要素の間には必ず R_{sd} の要素が含まれていなければならない．しかしながら $|R_{ds}| > |R_{sd}|$ であるから, ある二つの R_{ds} の要素間には R_{sd} が含まれておらず矛盾する． $|R_{sd}| > |R_{ds}|$ の場合も順序列における二つの R_{sd} の要素間には R_{ds} が含まれていなければならない, 同様に矛盾が起こる．

二つ目及び三つ目の場合, $S_i \in R_s$ または $S_i \in R_b$ であるような S_i に対し sv_i が RC に含まれるはずである．一方, $S_j \in R_d$ または $S_j \in R_b$ であるような S_j は, dv_j が RC に含まれるはずである．しかしながら $|R_{ds}| = |R_{sd}| = 0$ であるので補題 7.15 より, この二つの頂点を同時に含む有向サイクル C は存在しない．

次に, $|R_x| = 0$ かつ与えられた条件のいずれかを満たすような S の分割を持つ有向サイクル C が RC に含まれることを示す．それぞれの場合について考える．

(i) を満たす場合, $|R_{ds}| = |R_{sd}|$ である．ここで $R'_d = R_d \cup R_b$, $R'_s = R_s \cup R_b$ として定義する． $R'_d, R'_s, R_{ds}, R_{sd}$ の要素をそれぞれ $r'_d, r'_s, r_{ds}, r_{sd}$ とおき, 次のような順序列を考える．

$$r_{ds} (*r'_s) r_{sd} (*r'_d) r_{ds} \cdots r_{ds} (*r'_s) r_{sd} (*r'_d)$$

ただし, $(*r'_s), (*r'_d)$ とはそれぞれ R'_s, R'_d の要素を任意の数だけ並べたものとする．このとき, 隣り合う二つの要素は, 補題 7.15 より左の要素の葉から右の要素の根への弧が存在する．更に, 右端の R'_d あるいは R_{ds} の要素の葉から左端の R_{ds} の根への弧も存在するので, この列から C は RC に含まれる．

(ii) を満たす場合, 順序列 S_1, S_2, \dots, S_d について考える． S のすべての要素は R_d に属している．したがって, 補題 7.15 より順序列内の二つの要素 S_i, S_{i+1} ($1 \leq i \leq d-1$) に対し dv_i から dv_{i+1} への有向パス P_i が存在する．また, S_d から S_0 への有向パス P_{i+1} も同様に存在する．どの二つの有向パスの間にも重複する弧は存在しないため, これらの有向パスを繋げることで RC に含まれる有向サイクルが構成でき, これを root-cycle とする全域 cycle-rooted tree が存在する．

(iii) を満たす場合, (ii) の場合と同様に順序列 S_1, S_2, \dots, S_d について考える． S のすべての要素は R_s に属しており, sv_i から sv_{i+1} への有向パスがそれぞれの i に対して存在するため, それらを繋ぎ合わせることによって RC に含まれる有向サイクル及びそれを root-cycle

とする全域 cycle-rooted tree が構成できる . □

n が偶数の場合は次の定理が成り立つ .

定理 7.18 n を偶数とする . 有向サイクル C が RC に含まれるための必要十分条件は $|R_x| = 0$ であり , かつ次の条件のいずれかを満たすことである .

(i) $|R_d| = |R_s| \neq 0$.

(ii) $|R_{ds}| = d$.

(iii) $|R_{sd}| = d$.

$|R_d| = |R_s| = 0$ かつ ,

(iv) $|R_b| > 2$.

(v) $(|R_b| = 2) \wedge ((|R_{ds}| \geq 2) \vee (|R_{sd}| \geq 2))$.

(vi) $(|R_b| = 1) \wedge (|R_{ds}| \geq 1) \wedge (|R_{sd}| \geq 1)$.

証明 RC に含まれる有向サイクル C が上記の条件のいずれの条件も満たしていないと仮定する . このとき , 次の4つに場合分けすることができる .

- $|R_d| \neq |R_s|, |R_d| \neq d, |R_s| \neq d$.

$|R_d| = |R_s| = 0$ かつ ,

- $(|R_b| = 0) \wedge (|R_{ds}| > 0) \wedge (|R_{sd}| > 0)$.
- $(|R_b| = 2) \wedge (|R_{ds}| < 2) \wedge (|R_{sd}| < 2)$.
- $(|R_b| = 1) \wedge ((|R_{ds}| = 0) \vee (|R_{sd}| = 0))$.

いずれの場合にも矛盾が生じることを示す . まず , $R'_d = R_d \cup R_b, R'_s = R_s \cup R_b$ として定義する . $|R_d| \neq |R_s|, |R_d| \neq d, |R_s| \neq d$ の場合 , $|R_d| > |R_s|$ と仮定すると $|R'_d| > |R'_s|$ である . n は偶数であるから , 補題 7.16 より C 内で二つの R'_d の要素の間には必ず R'_s の要素が含まれていなければならない . しかしながら $|R'_d| > |R'_s|$ であるからある二つの R'_d の要素間には R'_s が含まれておらず , 矛盾が起こる . $|R_s| > |R_d|$ の場合も $|R'_s| > |R'_d|$ であるから二つの R'_s の要素間には R'_d が含まれていなければならない , 同様に矛盾が起こる .

$|R_d| = |R_s| = |R_b| = 0$ かつ $|R_{ds}| > 0, |R_{sd}| > 0$ の場合 , 補題 7.16 より C 内のある R_{ds} の要素とある R_{sd} の要素の間には必ず R'_d の要素が少なくとも一つ含まれていなければならない , $|R_d| = |R_b| = 0$ であることに矛盾する . 同様に , ある R_{sd} の要素とある R_{ds} の要素の間には必ず R'_s の要素が少なくとも一つ含まれていなければならない , $|R_s| = 0$ に矛盾する .

$|R_d| = |R_s| = 0, |R_b| = 2$ かつ $|R_{ds}| < 2, |R_{sd}| < 2$ の場合, 明らかに $2 \leq d \leq 4$ である. R_b に属する二つの S の要素を S_i, S_j とおく. このとき $|R_{ds}| < 2$ かつ $|R_{sd}| < 2$ であるから, RC が有向サイクルとなるには必ず S_i の葉から根への弧, あるいは S_j の葉から根への弧が存在しなければならないが, S_i, S_j が cycle-rooted tree であることからそのような弧は存在しない. したがってこのような C は存在しない.

$|R_d| = |R_s| = 0, |R_b| = 1$ かつ, $|R_{ds}| = 0$ または $|R_{sd}| = 0$ の場合, まず $|R_{ds}| = 0$ のとき R_b に属する S の要素を S_i とする. このとき C 内に $Co-sT_i^-$ の葉から dv_i への弧が存在しなければならず C の条件に矛盾する. 同様に $|R_{sd}| = 0$ のとき, C 内に $Co-sT_i^-$ の葉から dv_i への弧が存在しなければならず C の条件に矛盾する. したがって, どちらの場合も矛盾が生じる.

次に, $|R_x| = 0$ かつ与えられた条件のいずれかを満たすような S の分割を持つ有向サイクル C が RC に含まれていることを示す. それぞれの場合について考える.

(i) を満たす場合, $R'_d, R'_s, R_{ds}, R_{sd}$ の要素をそれぞれ $r'_d, r'_s, r_{ds}, r_{sd}$ とおき, 次のような順序列を考える.

$$r'_d (*r_{sd}) r'_s (*r_{ds}) r'_d \cdots r'_d (*r_{sd}) r'_s (*r_{ds}).$$

このとき, 隣り合う二つの要素は, 隣り合う r'_d と r'_s が同じ要素である場合を除いて補題 7.16 より左の要素の葉から右の要素の根への弧が存在し, 右端の R'_s あるいは R_{ds} の要素の葉から左端の R_d の根への弧も存在する. ここで, $|R_d| = |R_s| \neq 0$ であるからどの隣り合う r'_d と r'_s も異なる要素であるような並び方が存在し, この列を持つ有向サイクル C は RC に含まれる.

(ii) を満たす場合, 順序列 S_1, S_2, \dots, S_d について考える. S のすべての要素は R_{ds} に属している. したがって補題 7.16 より, 順序列内の二つの要素 S_i, S_{i+1} ($1 \leq i \leq d-1$) に対し dv_i から sv_{i+1} への有向パスが存在し, S_d から S_0 への有向パスも同様に存在する. どの二つの有向パスの間にも重複する弧は存在しないため, これらの有向パスを繋げることで RC に含まれる有向サイクルが構成でき, これを root-cycle とする全域 cycle-rooted tree が存在する.

(iii) を満たす場合, (ii) の場合と同様に順序列 S_1, S_2, \dots, S_d について考える. S のすべての要素は R_{sd} に属しており, sv_i から dv_{i+1} への有向パスがそれぞれの i に対して存在するため, それらを繋ぎ合わせることによって RC に含まれる有向サイクル及びそれを root-cycle とする全域 cycle-rooted tree が存在する.

(iv) を満たす場合についても同様に順序列を考える. $|R_b| > 2$ であるから, どの隣り合う r'_d と r'_s も異なる要素であるような並び方が存在するのでこの列を持つ有向サイクル C は RC に含まれる.

(v) を満たす場合も同様に R_b の要素を S_i, S_j とおくと, $|R_{ds}| \geq 2$ または $|R_{sd}| \geq 2$ であるから順序列は

$$S_i (*r_{sd}) S_j r_{ds} (*r_{ds}) S_j (*r_{sd}) S_i r_{ds} (*r_{ds}),$$

$$S_i r_{sd} (*r_{sd}) S_i (*r_{ds}) S_j r_{sd} (*r_{sd}) S_j (*r_{ds}),$$

のどちらかで表すことができる．したがって補題 7.16 よりこの列を持つ有向サイクル C は RC に含まれる．

(vi) を満たす場合, R_b の要素を S_i とおくと $|R_{ds}| \geq 1$ かつ $|R_{sd}| \geq 1$ であるから順序列は

$$S_i r_{sd} (*r_{sd}) S_i r_{ds} (*r_{ds}),$$

と表すことができる．したがって補題 7.16 よりこの列を持つ有向サイクル C は RC に含まれる． \square

図 7.6 に $K(3, 4)$ 上で構成可能な全域 cycle-rooted tree の例を示す．

7.2.2 全域 cycle-rooted tree を用いたアルゴリズム

この小節では Kautz ダイグラフにおける全域木及び前小節で構成した全域 cycle-rooted tree を用いたマルチソースブロードキャストについて述べる．Kautz ダイグラフ $K(d, n)$ 上でのマルチソースブロードキャストを行なう方法として, F_i から構成される全域木と, 前小節で構成した cycle-rooted tree を用いて考える．ここで与えられるアルゴリズムは高々 d 個の情報を同時に扱うことができる．

前小節で構成される全域 cycle-rooted tree S_i 上でのブロードキャストに必要なラウンド数は次で与えられる．

補題 7.19 $K(d, n)$ の部分ダイグラフ S_i に対して, dv_i をソース頂点としたときのブロードキャストに必要なラウンド数は $d(n-1) - 1$ である．

証明 dv_i から情報を送る順番として, まず $Co-dT_i^-$ のすべての子に情報を送信してから sv_i に情報を送るものとする．このとき $Co-dT_i^-$ 上でのブロードキャストに必要なラウンド数は $Co-dT_i^-$ が $CT^-(d, n-1)$ と同型であることと定理 7.2 から $d(n-1) - 1$ である．一方, $Co-sT_i^-$ 上でのブロードキャストに必要なラウンド数は sv_i に情報が送信されるのが d ラウンド目であること, $Co-sT_i^-$ が $CT^-(d, n-2)$ と同型であることから $d + d(n-2) - 1 = d(n-1) - 1$ である．したがって, S_i 全体でのブロードキャストに必要なラウンド数は $d(n-1) - 1$ である． \square

マルチソースブロードキャストを行なうための手順として, 最初にそれぞれの情報を特定の頂点に送信してから残りの頂点へすべての情報を送信することを考える．一つ目の方法として, 全域木を用いた $K(d, n)$ 上のマルチソースブロードキャストを考える．ここでは全域 cycle-rooted tree F_i から有向辺 (sv_i, dv_i) を取り除いた全域木を用いる．

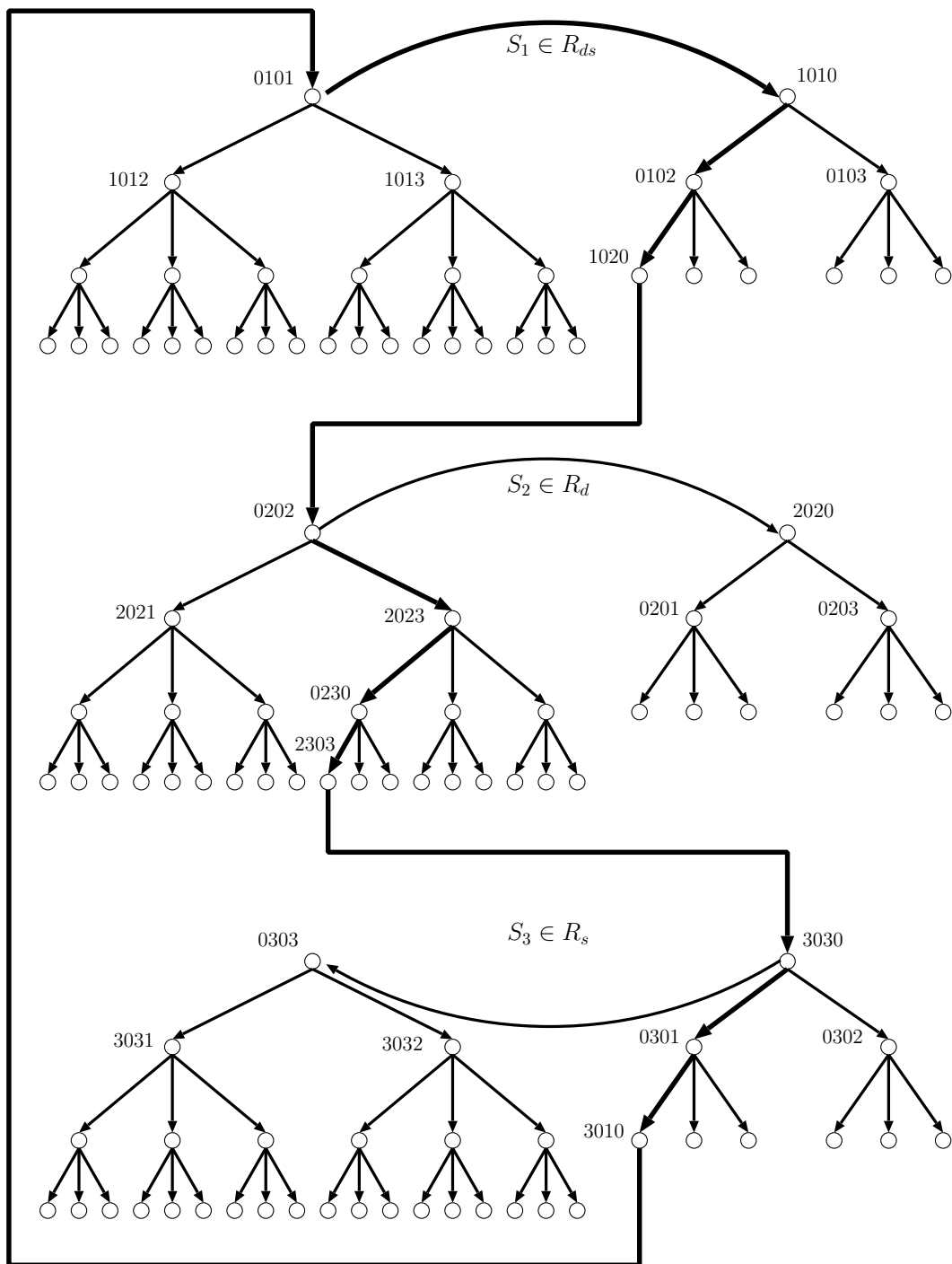


図 7.6: Kautz ダイグラフ $K(3,4)$ 上の全域 cycle-rooted tree の構成例

全域木上のすべての頂点に情報を送信するためには根から情報の送信を始めなければならず、前準備として各全域木に情報を送ることを考える。 $K(d, n)$ の直径は n であるから、任意の 2 頂点間の距離は高々 n である。また、各頂点は 1 ラウンドに高々一つの情報を受信し、別の一つの情報を送信することのみが行なえるため、 d 個の情報それぞれに対し全域木の根へ送信するまでに通る各頂点で起こる遅延の総和は高々 $d - 1$ である。したがって、 d 個の情報を全域木の根に送信するために必要なラウンド数は高々 $d + n - 1$ ラウンドであることがわかる。

これをもとに dv_i から他のすべての頂点にすべての情報を送信するために必要なラウンド数を考える。

定理 7.20 Kautz ダイグラフ $K(d, n)$ の全域木 F_i に対して、 dv_i から d 個の情報をすべての頂点に送信するために必要なラウンド数は高々 $d^2 + dn - d - 1$ である。

証明 dv_i が情報を送信する順番として、一つの情報をすべての子に送り終えてから次の情報を送信し始めるものとする。また、 dv_i の情報の送信順序としてまず sv_i を除くすべての子に送信してから、最後に sv_i に情報を送信するものとし、各頂点がそれぞれの情報を子に送る順番はすべて同じであるとする。

まず dv_i の collateral tree $Co-dT_i$ にすべての情報を送信するのに必要なラウンド数を考える。 $Co-dT_i$ 内で dv_i から隣接される各頂点に対して、 dv_i から一つ目の情報が送信されるのは i ($1 \leq i \leq d - 1$) ラウンド目であり、以降の情報は d ラウンドごとに送られる。今、これらの頂点を根とする部分木を考えると、各部分木は $CT(d, n - 1)$ と同型であり、葉以外のすべての頂点の出次数が d であることから情報の送受信の際に遅延は発生しない。したがって、各部分木がすべての情報を送信するのに必要なラウンド数は、最初の情報が送られるまでにかかるラウンド数 i と最後の情報が根に送られるまでにかかるラウンド数 $(d - 1)d$ 、および最後の情報を部分木のすべてに送信するために必要なラウンド数 $d(n - 1)$ の和 $d^2 + dn - 2d + i$ である。よって、最もラウンド数が遅れる部分木でも $d^2 + dn - d - 1$ ラウンドですべての情報を送信することができる。

次に $Co-sT_i$ にすべての情報を送信するのに必要なラウンド数を考える。 sv_i に最初の情報が送信されるのは d ラウンド目であり、次からの情報は d ラウンドごとに送られる。根と葉以外のすべての頂点の出次数は d である。さらに根の出次数は $d - 1$ であるから木の上ではどの頂点においても遅延が発生しない。また、 $Co-sT_i$ は $CT^-(d, n - 1)$ と同型であるから、定理 7.2 より $Co-sT_i$ 上でのブロードキャストに必要なラウンド数は $d(n - 1) - 1$ で与えられる。よって、 d 個の情報を $Co-sT_i$ のすべての頂点に送信するために必要なラウンド数は $d + d(d - 1) + d(n - 1) - 1 = d^2 + dn - d - 1$ となる。

以上より、 dv_i から d 個の情報をすべての頂点に送信するために必要なラウンド数は高々 $d^2 + dn - d - 1$ である。 \square

定理 7.20 より次も明らかとなる。

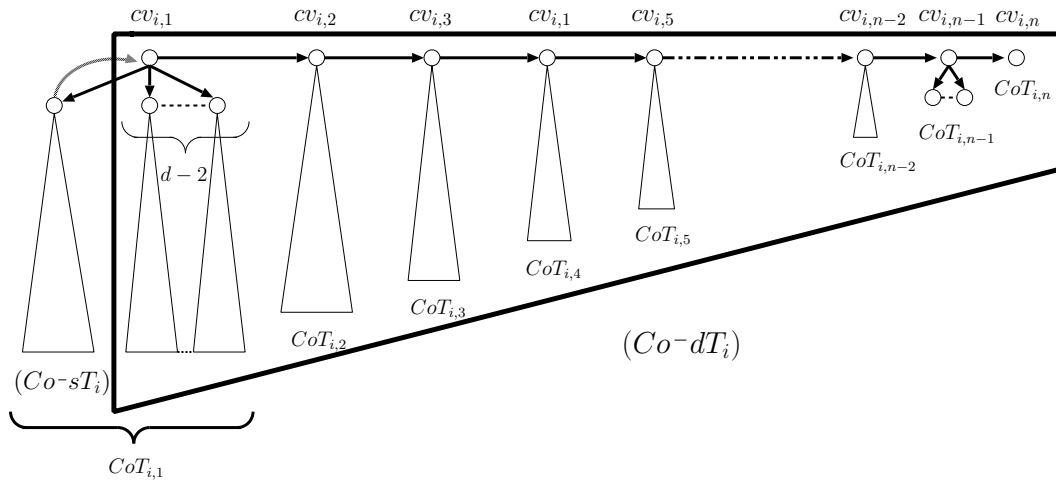


図 7.7: $S_i \in R_d$ の構造 (n が奇数の場合)

系 7.21 $d \geq 2, n \geq 2$ に対し Kautz グラフ $K(d, n)$ の高々 d 個のソースを持つマルチソースブロードキャストに必要なラウンド数は高々 $d^2 + dn + n - 2$ ラウンドである。

次に, cycle-rooted tree を用いてマルチソースブロードキャストを行なう方法を提案する。

$K(d, n)$ 上のマルチソースブロードキャストを行なう全域 cycle-rooted tree のモデルとして前小節で構成したものをを用いる。 n が奇数ならば S の各要素がすべて R_d に含まれる全域 cycle-rooted tree を用い, n が偶数ならばすべて R_{ds} に含まれる全域 cycle-rooted tree を用いることにする。また, root-cycle の繋がりとして S_i ($1 \leq i < d$) 上の有向パスの後には S_{i+1} 上の有向パスが繋がり, S_d 上の有向パスは S_1 上の有向パスへ繋がるものとする。以降, これらの全域 cycle-rooted tree を SPCRT と表記する。

SPCRT の root-cycle には n の偶奇によらず S_i 上での dv_i から葉への長さ n の有向パスが含まれる。今, SPCRT 上において有向パスの頂点を始点 dv_i から順に $cv_{i,1}, cv_{i,2}, \dots, cv_{i,n}$ と表す。 $1 \leq k < d$ に対し, $cv_{k,n}$ は $cv_{k+1,1}$ へ隣接し $cv_{d,n}$ は $cv_{1,1}$ へ隣接しているとする, SPCRT の root-cycle は

$$cv_{1,1}, cv_{1,2}, \dots, cv_{1,n}, cv_{2,1}, \dots, cv_{2,n}, \dots, cv_{d,1}, \dots, cv_{d,n}, cv_{1,1}$$

と表せる。 SPCRT 上での $cv_{i,j}$ の collateral tree をそれぞれ $CoT_{i,j}$ と表す。

それぞれの場合における各 S_i の cycle-vertex と collateral tree を図 7.7, 7.8 に示す。

全域 cycle-rooted tree を用いてマルチソースブロードキャストを効率よく行なうために, 前処理として各ソースの情報を root-cycle 上に送信し, 情報をバランスよく再配置する必要がある。この前処理に関して次が成り立つ。

補題 7.22 d 個の情報を各 dv_i に一つずつ送信するために必要なラウンド数は高々 $n + 1$ である。

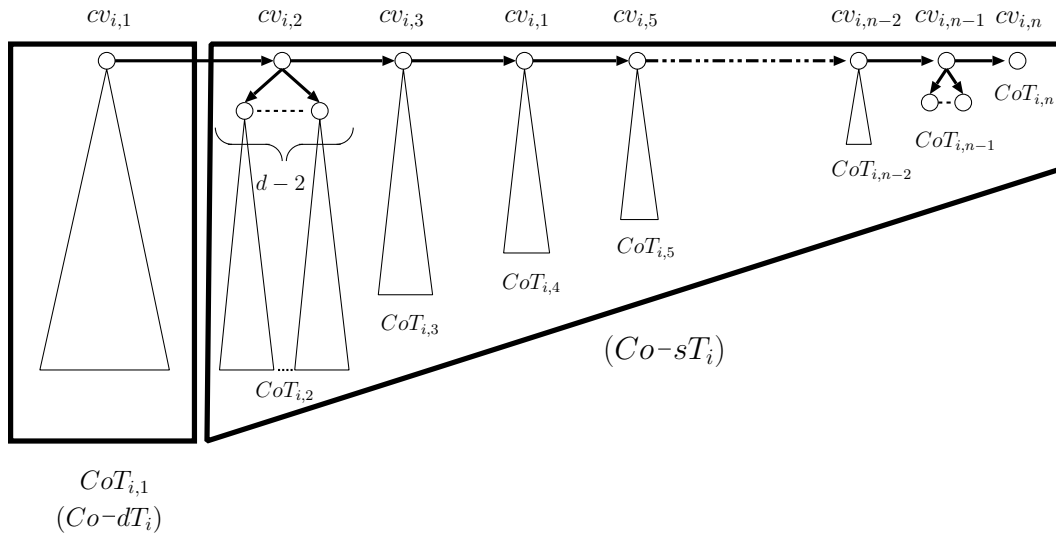


図 7.8: $S_i \in R_{ds}$ の構造 (n が偶数の場合)

証明 d 個の情報をそれぞれ m_1, m_2, \dots, m_d とおき, それぞれの情報を持っているソース頂点を s_1, s_2, \dots, s_d とおく. また, s_i ($1 \leq i \leq d$) から dv_i への次のような有向パスを P_i とおく.

$s_i = (x_0, x_1, \dots, x_{n-1})$ に対し,

1. $x_{n-1} = i$ ならば s_i から dv_i への最短有向パス.
2. $x_{n-1} \neq i$ ならば 有向辺 (s_i, s'_i) ($s'_i = x_1, \dots, x_{n-1}, i$) を通った後, s'_i から dv_i への最短有向パス.

上記の手順から作られる有向パスにしたがって各情報を送信することを考える.

s_i を除く P_i 上の頂点 $v = (v_0, v_1, \dots, v_{n-1})$ はすべて $v_{n-1} = i$ または $v_{n-2} = i, v_{n-1} = 0$ を満たす. したがって, 二つの有向パス P_i, P_j における頂点の共通部分として s_i あるいは s_j のみが考えられる. しかしながら s_i が m_i を送信するのが 1 ラウンド目であるのに対して, s_i が m_j を受信するのは 2 ラウンド目以降である. したがって各情報を送信する際に遅延は起こらず, P_i の長さは n が奇数の場合は高々 n , n が偶数の場合は高々 $n+1$ であることから命題が成り立つ. \square

補題 7.22 を用いて前処理を行なった後で, dv_i から情報の送信を新たに始めるものとする. このとき, $SPCRT$ 上でのマルチソースブロードキャストを行うために必要なラウンド数は次で与えられる.

補題 7.23 各 dv_i が高々一つ情報を持っているとする. このとき $SPCRT$ 上でのマルチソースブロードキャストに必要なラウンド数は $d < n$ の場合高々 $2dn - d - n - 1$ であり, $d \geq n$ の場合高々 $d^2 + dn - 2d$ である.

証明 葉以外のすべての頂点が情報を送信する順序の規則として、情報 m_i, m_j をこの順番で受信したときすべての子に m_i を送信してから m_j を送信し始めること、自身が cycle-vertex であるならばまず隣接する cycle-vertex に情報を送信することを定める。また、 dv_i が子に情報を送信する順序として sv_i が cycle-vertex でない限り必ず最後となるようにする。この送信規則において、各情報を dv_i が送信してから次の dv_j が受信するまでにかかるラウンド数は n の偶奇によらず n となる。

今、 S の各要素は n が奇数の場合はすべて R_d に属し、 n が偶数の場合はすべて R_{ds} に属している。したがって、全域 cycle-rooted tree 内での各要素の構造は互いに同型である。このことから、ある S の一つの要素 S_i のすべての頂点へすべての情報を送信するために必要なラウンド数が得られれば、それが $K(d, n)$ のマルチソースブロードキャストに要するラウンド数となる。これを $d < n$ と $d \geq n$ の二つの場合に分けて求める。

dv_i がすべての隣接頂点に一つの情報を送信するためには d ラウンドを必要とする。 $d < n$ の場合、 dv_i がある情報を受信するまでにその前に受信した情報をすべての子に送信し終えていることは明らかである。よって dv_i における遅延は発生せず、残りの頂点も遅延が発生しないことから S_i 内での遅延は発生しない。

したがって S_i のすべての頂点にすべての情報を送信するのに必要なラウンド数は d 個目の情報を dv_i が受信するまでにかかるラウンド数 $(d-1)n$ と d 個目の情報を S_i のすべての頂点に送信するのにかかるラウンド数の和である。 n が奇数ならば dv_i の子の中で sv_i に情報が送信されるのは最後となるため、 d 個目の情報をすべての頂点に送信するのに必要なラウンド数は補題 7.19 より $d(n-1) - 1$ ラウンドである。一方、 n が偶数である場合 sv_i が cycle-vertex であることから送信規則により dv_i の子の中で最初に sv_i に情報が送信される。これにより d 個目の情報を S_i のすべての頂点に送る際に d 番目に $CT(d, n-2)$ の根である dv_i の子に情報を送信することとなり、これに必要なラウンド数は $dn - d$ となる。したがって S_i のすべての頂点にすべての情報を送信するために必要なラウンド数は高々 $(d-1)n + dn - d = 2dn - d - n$ となる。

$d \geq n$ の場合、 dv_i がある情報をすべての子に送信し終える前に次の情報を受信してしまう。すなわち、 dv_i がある情報をすべての子に送信し終えたときに必ず次の情報をすでに受信しているということを意味する。

したがって S_i のすべての頂点にすべての情報を送信するのに必要なラウンド数は $d-1$ 個の情報をすべての子に送信するためにかかるラウンド数 $(d-1)d$ と d 個目の情報が S_i のすべての頂点にかかるラウンド数の和となる。よって $d \geq n$ の場合に必要なラウンド数は高々 $d(d-1) + dn - d = d^2 + dn - 2d$ となる。□

補題 7.22, 7.23 より次が従う。

定理 7.24 $d \geq 2, n \geq 2$ に対し Kautz ダイグラフ $K(d, n)$ のソースが高々 d 個のマルチソースブロードキャストに必要なラウンド数は $d < n$ のとき高々 $2dn - d$ ラウンドであ

り, $d \geq n$ のとき高々 $d^2 + dn - 2d + n + 1$ ラウンドである.

7.2.3 SPCRT に用いられない弧の性質

ここでは $K(d, n)$ 上でのマルチソースブロードキャストを行うために用いられた SPCRT の性質についての考察を行なう. SPCRT は $K(d, n)$ の部分グラフであり, 各 collateral tree の葉以外の頂点については出次数が d であるから $K(d, n)$ 上でこれらの頂点から出ている弧はすべて SPCRT 上にも現れる. しかし, collateral tree の葉については出次数が 0 であり, これらの頂点から出ている弧はすべて SPCRT 上に現れることはない. これらのことをもとに SPCRT 上に現れない $K(d, n)$ の弧が実際にはどのように SPCRT 上に分布しているかについて考察する.

n が奇数か偶数かによって SPCRT 中に各 S_i が表れる形は異なっている. しかしながら, どちらの場合にも図 7.7, 図 7.8 で示されているように各 S_i 上で葉であった頂点は $cv_{i,n}$ を除いて SPCRT 上においても葉であるので, それぞれの場合に対して S_i の葉および $cv_{i,n}$ がどのような頂点へと隣接し, それらが SPCRT 上でどのような構造をしているかを示す.

n が奇数のとき, 各 S_i は R_d に属している. まず $CoT_{i,1}$ の葉から出る弧について考える. $CoT_{i,1}$ の根 $cv_{i,1}$ は $(i, 0, \dots, 0, i)$ であり, $CoT_{i,1}$ 上の子として $(0, \dots, i, j)$, ($j \neq i + 1$) で表される頂点を持ち, それらの一つとして $sv_i = (0, i, \dots, i, 0)$ が含まれている. $Co-sT_i$ は $CT^-(d, n - 2)$ と同型であるから, その葉は $(i, 0, k, *, \dots, *)$, ($k \neq i$) と表される. これらの葉は $K(d, n)$ 上で $(0, k, *, \dots, *)$ と表される頂点への弧を持っており, これは各 S_k 上で dv_k からの距離が奇数である頂点に対応している. $cv_{i,1}$ の sv_i 以外の子は $(0, i, \dots, 0, i, l)$, ($l \neq 0, i + 1$) と表され, これを根とする部分木の葉は $(i, l, *, \dots, *)$ であり, これらの葉が $K(d, n)$ 上で隣接している頂点は $(l, *, \dots, *)$ と表され, これらは S_l 上で dv_l からの距離が偶数である頂点に対応している.

次に $CoT_{i,p}$, ($1 < p \leq n$) の葉から出る弧について考える. SPCRT の構成により o を $1 < o \leq n$ を満たす奇数とすると各 $cv_{i,o}$ は

$$cv_{i,o} = \underbrace{(i, 0, \dots, 0, i)}_{n-o+1}, \underbrace{(i+1, 0, \dots, (i+1), 0)}_{o-1}$$

とラベル付けされており, e を $1 < e \leq n$ を満たす偶数とすると

$$cv_{i,e} = \underbrace{(0, i, \dots, 0, i)}_{n-e+1}, \underbrace{(i+1, 0, \dots, 0, (i+1))}_{e-1}$$

とラベル付けされている. これらのことから, SPCRT 上において左 1 桁が i である頂点は $CoT_{i,o}$ の高さが偶数の場所, または, $CoT_{i,e}$ の高さが奇数の場所に属しているということがいえる. 同様に左 2 桁が $0i$ である頂点については $CoT_{i,o}$ の高さが奇数の場所, または, $CoT_{i,e}$ の高さが偶数の場所に属していることもいえる. $CoT_{i,p}$ ($1 < p \leq n$) は $CT^-(d, n - p)$

と同型であるので, $CoT_{i,1}$ 以外の collateral tree の葉はすべて $(i, (i+1), *, \dots, *)$ と表される. これらの葉は $K(d, n)$ 上で $((i+1), *, \dots, *)$ で表される頂点へ隣接し, 補題 7.14 より隣接先はすべて S_{i+1} 上の頂点となる.

さらに詳しく隣接関係を調べると, $CoT_{i,o}$ の葉は

$$(i, \underbrace{(i+1), 0, \dots, (i+1), 0}_{o-1}, \underbrace{j, *, \dots, *}_{n-o})$$

で表され, S_{i+1} 上の

$$(\underbrace{(i+1), 0, \dots, (i+1), 0}_{o-1}, \underbrace{j, *, \dots, *}_{n-o+1})$$

と表される頂点へ隣接している ($j \neq i+1$). 今, これらの頂点が $Co-dT_{i+1}$ に含まれていないことに注意すると, SPCRT 上で $CoT_{i+1,1}$ の部分木 $Co-sT_{i+1}$ の中で高さが $n-o+1$ の場所に上記の頂点が存在していることがわかる.

$CoT_{i,e}$ の葉は

$$(i, \underbrace{(i+1), 0, \dots, 0, (i+1)}_{e-1}, \underbrace{k, *, \dots, *}_{n-e})$$

で表され ($k \neq 0$), S_{i+1} 上の

$$(\underbrace{(i+1), 0, \dots, 0, (i+1)}_{e-1}, \underbrace{k, *, \dots, *}_{n-e+1})$$

で表される頂点へと隣接している. SPCRT 上においてこれらのラベルに対応する頂点は $CoT_{i+1,q}$ の根

$$(\underbrace{(0 \text{ or } (i+1)), \dots, 0, (i+1)}_{n-q+1}, \underbrace{(i+2), 0, *, \dots, (0 \text{ or } (i+2))}_{q-1})$$

から高さ $n-q+1-(e-1) = n-q-e+2$ の場所に存在する. ただし, 高さは 0 以上であるため $q \leq n-e+2$ の場合の collateral tree に限る.

n が奇数のとき SPCRT 上で各 collateral tree の葉から出ている弧の様子を図 7.9 に示す.

n が偶数のとき, 各 S_i は R_{ds} に属している. 図 7.8 のように R_{ds} に属している要素は $CoT_{i,1}$ が $Co-dT_i$ そのものであり, 残りの collateral tree はそれぞれ $Co-sT_i$ の部分木となっている. まず $CoT_{i,1}$ の葉について考えると, $Co-dT_i$ の構造から各葉は $(i, j, *, \dots, *)$, ($j \neq 0$) と表される. これらの頂点は, $K(d, n)$ 上で $(j, *, \dots, *)$ と表される頂点へ隣接している. 補題 7.14 より, これらは S_j に属する dv_j から距離が奇数の頂点に対応している. $CoT_{i,2}$ について, $cv_{i,2} = (i, 0, \dots, i, 0)$ は $(0, i, \dots, i, 0, k)$, ($k \neq i, i+1$) で表される $d-2$ 個の子を持っており, 各子を根とする部分木 $CT(d, n-3)$ の葉は $(i, 0, k, *, \dots, *)$ で表され, これらの葉は $(0, k, *, \dots, *)$ なる頂点へ隣接している. 補題 7.14 から, 隣接先の頂点はそれぞれ S_k に属しており, S_k 上で dv_k から距離が偶数の位置に存在している.

次に, $CoT_{i,p}$, ($2 < p \leq n$) の葉から出る弧について考える. SPCRT の構成により, o を $2 < o \leq n$ を満たす奇数とすると各 $cv_{i,o}$ は

$$cv_{i,o} = (\underbrace{0, i, \dots, 0, i}_{n-o+1}, \underbrace{0, (i+1), \dots, 0, (i+1)}_{o-1})$$

とラベル付けされており, e を $2 < e \leq n$ を満たす偶数とすると

$$cv_{i,e} = (\underbrace{i, 0, \dots, 0, i}_{n-e+1}, \underbrace{0, (i+1), \dots, (i+1), 0}_{e-1})$$

とラベル付けされている. これらのことから, SPCRT 上において左端の桁が i である頂点は $CoT_{i,o}$ の高さが奇数の場所, または $CoT_{i,e}$ の高さが偶数の場所に属しているということがいえる. 同様に左端から 2 桁が $0i$ である頂点については $CoT_{i,o}$ の高さが偶数の場所, または $CoT_{i,e}$ の高さが奇数の場所に属していることもいえる. $CoT_{i,p}$ ($1 < p \leq n$) は $CT^-(d, n-p)$ と同型であるので, $CoT_{i,p}$ の葉はすべて $(i, 0, (i+1), *, \dots, *)$ と表される. これらの葉は $K(d, n)$ 上で $(0, (i+1), *, \dots, *)$ で表される頂点へ隣接し, 補題 7.14 より隣接先はすべて S_{i+1} 上の頂点となる.

さらに詳しく隣接関係を調べると, $CoT_{i,o}$ の葉は

$$(i, \underbrace{0, (i+1), \dots, 0, (i+1)}_{o-1}, \underbrace{k, *, \dots, *}_{n-o})$$

で表され ($k \neq 0$), S_{i+1} 上の

$$(\underbrace{0, (i+1), \dots, 0, (i+1)}_{o-1}, \underbrace{k, *, \dots, *}_{n-o+1})$$

と表される頂点へ隣接している. 今, これらの頂点はすべて $Co-dT_{i+1}$ に含まれていることに注意すると, SPCRT 上で $CoT_{i+1,1}$ の部分木 $Co-sT_{i+1}$ の中で, 高さが $n-o+1$ の場所に上記の頂点が存在していることがわかる.

$CoT_{i,e}$ の葉は

$$(i, \underbrace{0, (i+1), \dots, (i+1), 0}_{e-1}, \underbrace{j, *, \dots, *}_{n-e})$$

で表され ($j \neq i+1$), S_{i+1} 上の

$$(\underbrace{0, (i+1), \dots, (i+1), 0}_{e-1}, \underbrace{j, *, \dots, *}_{n-e+1})$$

で表される頂点へと隣接している. SPCRT 上において, これらのラベルに対応する頂点は $CoT_{i+1,q}$ ($2 \leq q \leq n$) の根

$$(\underbrace{(0 \text{ or } (i+1)), \dots, (i+1), 0}_{n-q+2}, \underbrace{(i+2), *, \dots, (0 \text{ or } (i+2))}_{q-2})$$

から高さ $n - q + 2 - (e - 1) = n - q - e + 3$ の場所に存在する．ただし，高さは 0 以上であるため $q \leq n - e + 3$ の場合の collateral tree に限る．

n が偶数のとき SPCRT 上で各 collateral tree の葉から出ている弧の様子を図 7.10 に示す．

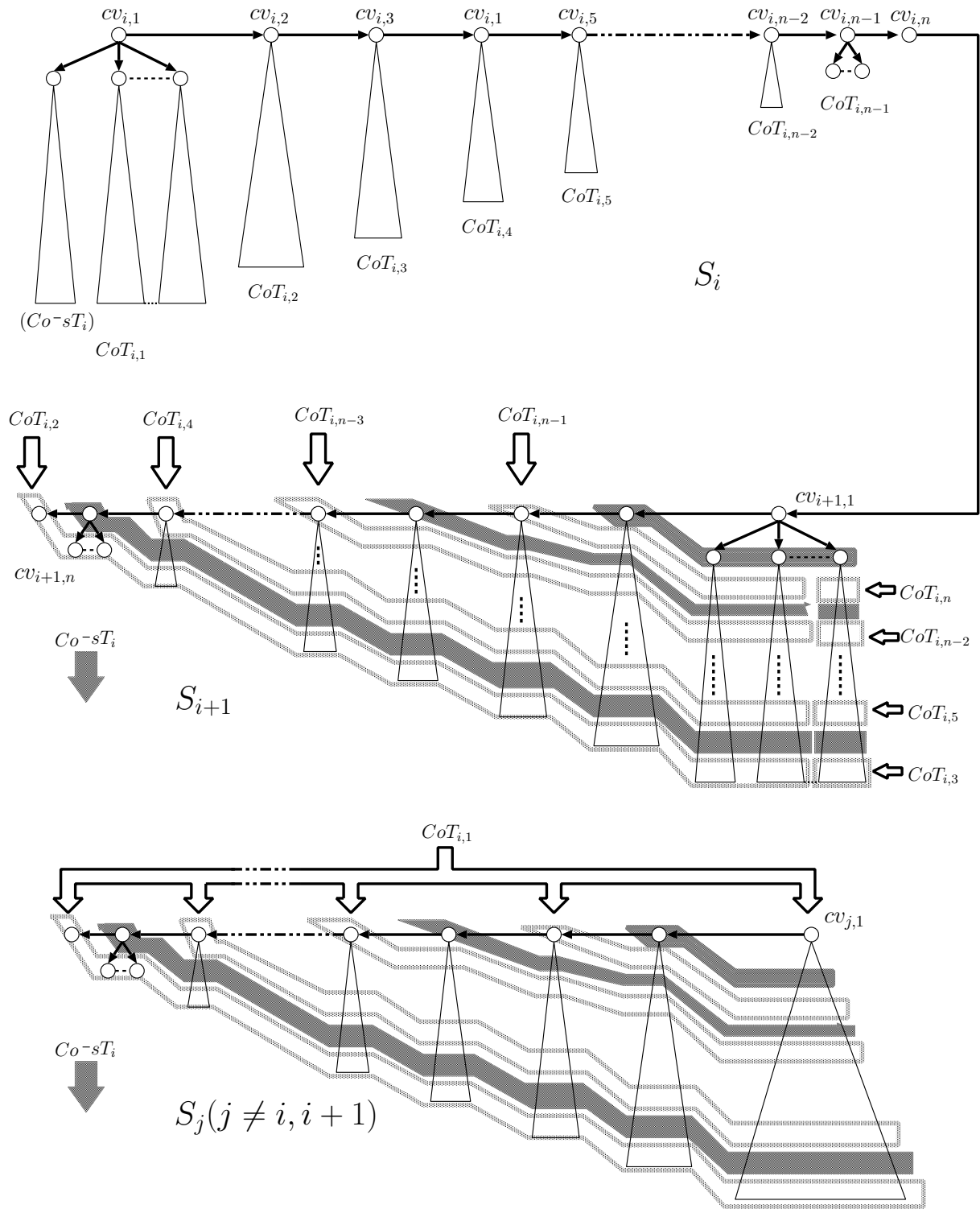


図 7.9: SPCRT の各葉が隣接する頂点 (n が奇数の場合)

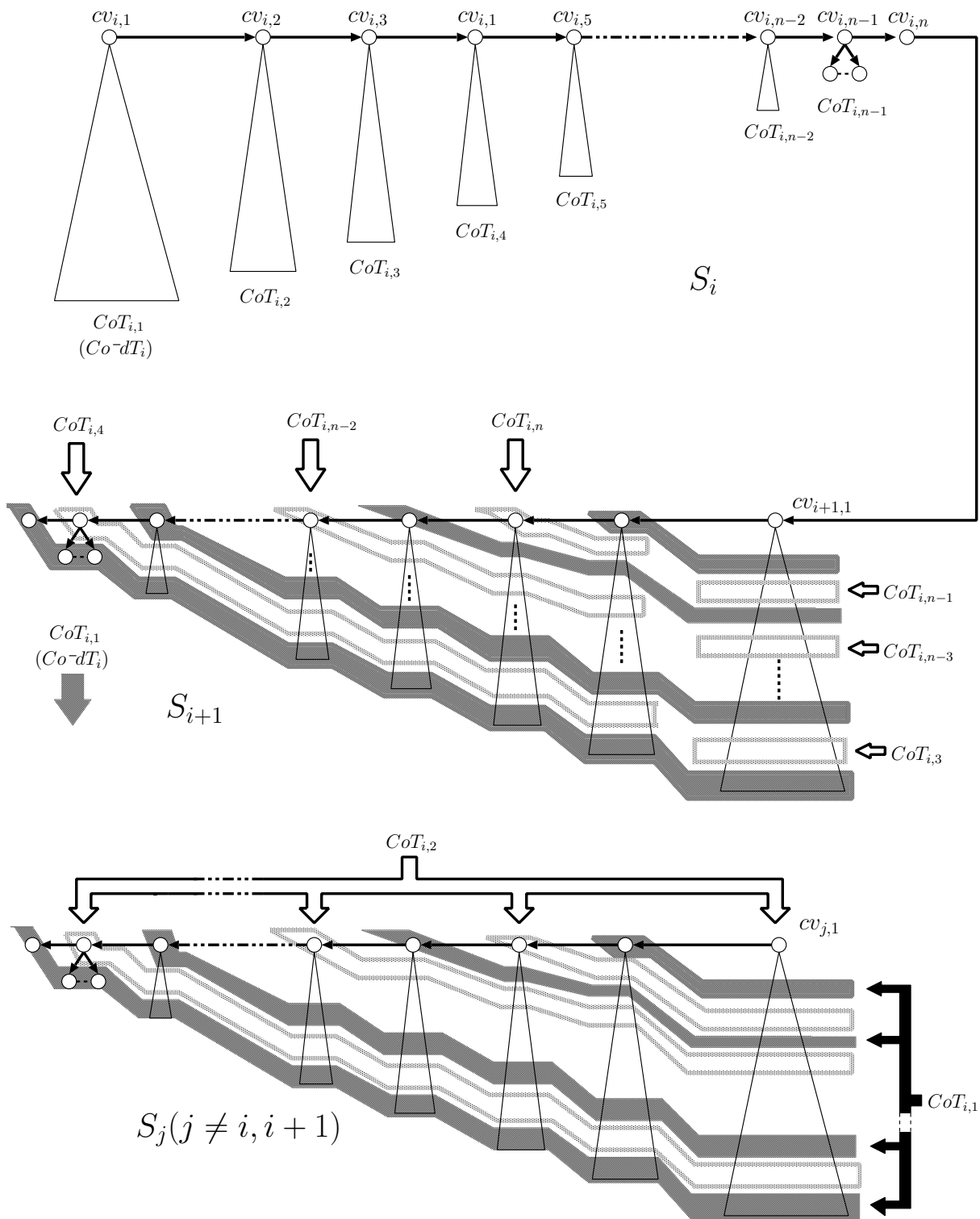


図 7.10: SPCRT の各葉が隣接する頂点 (n が偶数の場合)

7.2.4 cycle-rooted tree による分解を用いたアルゴリズム

本小節では Kautz ダイグラフ $K(d, n)$ 上でのマルチソースブロードキャストアルゴリズムを与える。まず, 7.1.2 小節で de Bruijn ダイグラフに用いたものと同様に, $K(d, n)$ 上で高々 d 個の情報を同時に扱うアルゴリズムを与える。

$B(d, n)$ に対しては完全 d 進 loop-rooted tree $SPCRT_i$ による同型因子分解を用いたが, $K(d, n)$ に対しては 2-cycle-rooted tree による同型因子分解, すなわち, 4.2.2 小節で定義した F_i を用いる。 $K(d, n)$ はループ頂点を持っておらず, F_i による同型因子分解は $K(d, n)$ 上でもっとも小さな root-cycle を持つ cycle-rooted tree によるものと考えられる。

$K(d, n)$ の頂点集合についても次のような分割を考える。

$$V(y_1, y_2, \dots, y_{n-1}) = \{(y_1, \dots, y_{n-1}, \beta) \mid 0 \leq \beta \leq d, y_{n-1} \neq \beta\}, \\ 0 \leq y_i \leq d, 0 \leq i \leq n-1.$$

このとき, de Bruijn ダイグラフの場合と同様に次が成り立つ。

補題 7.25 $K(d, n)$ に対し $V(y_1, y_2, \dots, y_{d-1})$ に属する d 個の頂点はすべて同じ頂点から隣接されている。

さらに, $|V(y_1, y_2, \dots, y_{d-1})| = d$ であるから整数集合 $\mathbb{Z}_d = \{1, 2, \dots, d\}$ への前小節で定義された d 個の bijection L_1, \dots, L_{d-1} が存在する。

各 F_i 上でブロードキャストを行なうためには, 二つの cycle-vertex のどちらかが情報を持っている必要がある。ここで F_i は d 正則であり, dv_i を根とする木の深さは n であり, sv_i を根とする木の深さは $n-1$ であるため, dv_i が情報を持っている場合の方がブロードキャストにかかる時間は少ないことが明らかである。 $K(d, n)$ の部分ダイグラフ F_i は $K(d, n+1)$ の部分ダイグラフ S_i と同型であるから, dv_i からのブロードキャストに必要なラウンド数は補題 7.19 より $dn-1$ で与えられる。また, 各 dv_i に一つずつ情報を送信するために必要なラウンド数は補題 7.22 より $n+1$ である。

de Bruijn ダイグラフの場合と同様に各 F_i に対し補題 7.6 の条件を満たす bijection を与える。ただし, 用いる bijection はその中で $0 \leq i \leq d-1$ に対して $L_i(v_{i0}) = d$ および $L_i(v_{0i}) = d$ を満たすものとする。このとき次が成り立つ。

補題 7.26 各 F_i 上でのブロードキャストに対して頂点 v が情報 m_i を受け取るラウンドを $r_i(v)$ とおく。このとき, bijection L_i が頂点 dv_i および sv_i に対して $L_i(dv_i) = L_i(sv_i) = d$ を満たすならば, 総ラウンド数が $dn-1$ であり任意の頂点 v に対して $L_i(v) = (r_i(v) \bmod d)$ であるような F_i 上のブロードキャストアルゴリズムが存在する。

今 $K(d, n)$ の二つの異なる F_i, F_j ($0 \leq i, j \leq d-1$) と任意の頂点 v に対し $r_i(v) \neq r_j(v)$ が成り立つ。したがって, 任意の異なる二つの F_i 上においてどの頂点もそれぞれ異なるラ

ウンドで情報を得ることがいえ，これは与えられた方法で各 F_i 上で同時にブロードキャストを行う際にどの頂点も同じラウンドに二つ以上の情報を受信することがないということを示している．

定理 7.27 $K(d, n)$ に対し，各ループ頂点が一つずつ異なる情報を持っていると仮定する．このとき $K(d, n)$ のすべての頂点に d 個すべての情報を送信するために必要なラウンド数は高々 $dn - 1$ である．

さらに，定理 7.22, 定理 7.27 より高々 d 個の情報をマルチソースブロードキャストするには高々 $(d + 1)n$ ラウンド必要であることがわかる．したがって $K(d, n)$ 上でマルチソースブロードキャストを行うために必要なラウンド数は次のようになる．

定理 7.28 $X = \{x_1, x_2, \dots, x_k\} \subset V(K(d, n))$, $1 \leq k \leq d^n + d^{n-1}$ を $K(d, n)$ 上のソース頂点とする．このとき $K(d, n)$ 上で k 個の情報をマルチソースブロードキャストするために必要なラウンド数は高々 $\lceil k/d \rceil (d + 1)n$ である．

本章では最初に de Bruijn ダイグラフ $B(d, n)$ 上での loop-rooted tree によるマルチソースブロードキャストアルゴリズムを与えた．これに対し，同様の方法を $K(d, n)$ 上の 2-cycle-rooted tree へと適用した． $K(d, n)$ はループを持たないため， $K(d, n)$ が含む cycle-rooted tree でもっとも小さな root-cycle の長さは 2 である．そのため，cycle-vertex の総数は $B(d, n)$ の場合よりも多く，これは同時に扱える情報の総数も同様に増えることがいえる．

本小節の残りではより多くの情報を同時に扱うような $K(d, n)$ 上でのマルチソースブロードキャストアルゴリズムを与える．

まず $K(d, n)$ に対して，次のような分解を与える．これは定理 4.23 の $h = 2$ である特別な場合として与えられる．

補題 7.29 Kautz ダイグラフ $K(d, n)$ は $\binom{d+1}{2}$ 個の高さ $n - 1$ の完全 d 進 2-cycle-rooted tree によって分解可能である．

証明 次のような $K(d, n)$ の弧部分集合を考える．

$$A_{ij} = \{((x_0, x_1, \dots, x_{n-1}), (x_1, x_2, \dots, x_n, x_n) \mid x_0, x_1 \in \{i, j\}, x_0 \neq x_1\}, \\ 0 \leq i < j \leq d.$$

$i < j$ のとき， A_{ij} なる弧部分集合は存在するが， A_{ji} なる弧部分集合は存在しないことに注意する． $0 \leq i < j \leq d$ であるから， A_{ij} の総数は

$$1 + 2 + \dots + d = \frac{d(d+1)}{2} = \binom{d+1}{2},$$

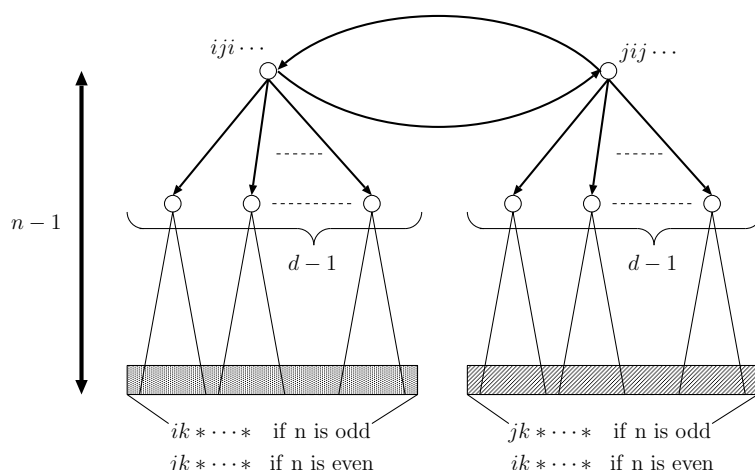


図 7.11: $K(d, n)$ の完全 d 進 2-cycle rooted tree による分解

である．各 A_{ij} は左端の 2 桁が i と j である頂点から接続している全ての弧からなので，二つの異なる部分集合は明らかに弧素である．さらに， $|A_{ij}| = d \cdot 2d^{n-2}$ であるので，

$$\sum_{0 \leq i < j \leq d} |A_{ij}| = \frac{d(d+1)}{2} \cdot 2d \cdot d^{n-1} = d(d^n + d^{n-1}),$$

が得られ，これは $K(d, n)$ の弧の総数に等しい．したがって $K(d, n)$ が A_{ij} により分解可能なことが示せた．

次に，各 $\langle A_{ij} \rangle$ が高さ $n-1$ の完全 d 進 2-cycle-rooted tree であることを示す．定義より， A_{ij} は i と j の交互列のみからなる二つの頂点を持ち，これらは双方向の弧を持つ．よって $\langle A_{ij} \rangle$ は長さ 2 の有向サイクルを持つ． $\langle A_{ij} \rangle$ に含まれる頂点は左端の桁が i か j であるから，上記の有向サイクルの頂点から他の任意の頂点への有向パスが存在し， $\langle A_{ij} \rangle$ は弱連結である．さらに，頂点 $v = (v_0, v_1, \dots, v_{n-1})$ は $v_0 = i$ または j であるから，部分ダイグラフ内で隣接される頂点は $(i, v_0, v_1, \dots, v_{n-2})$ と $(j, v_0, v_1, \dots, v_{n-2})$ のどちらかのみである．よって， $\langle A_{ij} \rangle$ は 1 入正則であるから 2-cycle-rooted tree であることがいえる．また，葉以外の頂点は d 個の頂点に隣接しており，root-cycle からの各葉までの距離が高々 $n-1$ であるから $\langle A_{ij} \rangle$ は高さ $n-1$ の完全 d 進 2-cycle-rooted tree である． \square

補題 7.29 により得られる 2-cycle-rooted tree を図 7.11 に示す．

$\langle A_{ij} \rangle$ は $(i, *, \dots, *)$ または $(j, *, \dots, *)$ で表されるすべての頂点を頂点集合として持つ．また，すべての葉は $(i, k, *, \dots, *)$ または $(j, k, *, \dots, *)$ ($k \neq i, j$) で表される．

今， $K(d, n)$ が持つ情報の総数は高々 $d(d-1)$ 個であるとする．このとき， $K(d, n)$ 上でのマルチソースブロードキャストを行なうために次のような情報散布の方法を与える．

phase 1.

2-cycle 上のすべての頂点に一つずつ情報を送信；

phase 2.

補題 7.29 より得られる誘導部分ダイグラフ上での情報散布;

phase 3.

各隣接頂点に必要な情報を送信;

上で定めた各 phase 毎に必要なラウンド数を求めることで $d(d+1)$ 個の情報のマルチソースブロードキャストに必要なラウンド数が得られる。まず phase 1. について考えると、必要なラウンド数の上界は以下で与えられる。

補題 7.30 phase 1. に必要なラウンド数は高々 $(d+1)n$ ラウンドである。

証明 まず $K(d, n)$ 上の d 個の cycle-vertex $(i, 0, i, 0, \dots)$, $(0 \leq i \leq d, i \neq 0)$ にそれぞれ一つずつ互いに異なる情報を同時に送信することを考える。 d 個の情報をそれぞれ m_1, m_2, \dots, m_d とおき、それぞれの情報を持っているソース頂点を s_1, s_2, \dots, s_d とおく。ここで、 $s_i = (x_1, x_2, \dots, x_n)$ $(1 \leq i \leq d)$ から頂点 $(i, 0, i, 0, \dots)$ への最短の有向パスを P_i とおくと、 s_i を除く P_i の頂点はすべて (\dots, i) または $(\dots, i, 0)$ で表せる。したがって、任意の二つの有向パス P_i, P_j における頂点の共通部分は各ソース頂点のみが考えられる。さらに、 s_i を含む P_i 以外の最短有向パス P_j は $j = x_n$ を満たさなければならず、そのようなソース頂点は高々一つしか存在しない。もし上の条件を満たす s_j が存在しても、 s_i が m_i を送信するのが1ラウンド目であり、 s_i が m_j を送信するのは2ラウンド目以降である。したがって、この方法により d 個の情報を送信する際に各頂点と同ラウンド中に二つ以上情報を受信することはない。 $\text{diam } K(d, n) = n$ であるから、 P_i の長さは高々 n であるから、これに必要なラウンド数は高々 n ラウンドである。 $K(d, n)$ 上の残りの d^2 個の情報に対しても、上と同様の方法で d 個の情報を d 個の頂点 (i, x, i, x, \dots) , $(1 \leq x \leq d, i \neq x)$ へ送信すると、必要なラウンド数は dn で与えられる。以上より、 $d(d+1)$ 個の情報を $d(d+1)$ 個のソース頂点にそれぞれ一つずつ送信するために必要なラウンド数は高々 $(d+1)n$ ラウンドである。 \square

次に phase 2. について考える。準備として、 $K(d, n)$ の頂点集合を次のように分割する。

$$V(v_0, v_1, \dots, v_{n-2}) = \{(v_0, v_1, \dots, v_{n-2}, \alpha) \mid \alpha = 0, 1, \dots, d, \alpha \neq v_{n-2}\}, \\ 0 \leq x_i \leq d, 1 \leq i \leq n-1.$$

このとき、各 $V(v_0, v_1, \dots, v_{n-2})$ に含まれる頂点はすべて同じ頂点から隣接されていることは明らかである。よって、 $V(v_0, v_1, \dots, v_{n-2})$ から整数集合 $\mathbb{Z}_d = \{0, 1, \dots, d-1\}$ への bijection について考えると、次が成り立つ。

補題 7.31 任意の $V(v_0, v_1, \dots, v_{n-2})$ に対し、次の条件を満たすような \mathbb{Z}_d への d 個の bijection L_0, L_1, \dots, L_{d-1} が存在する。

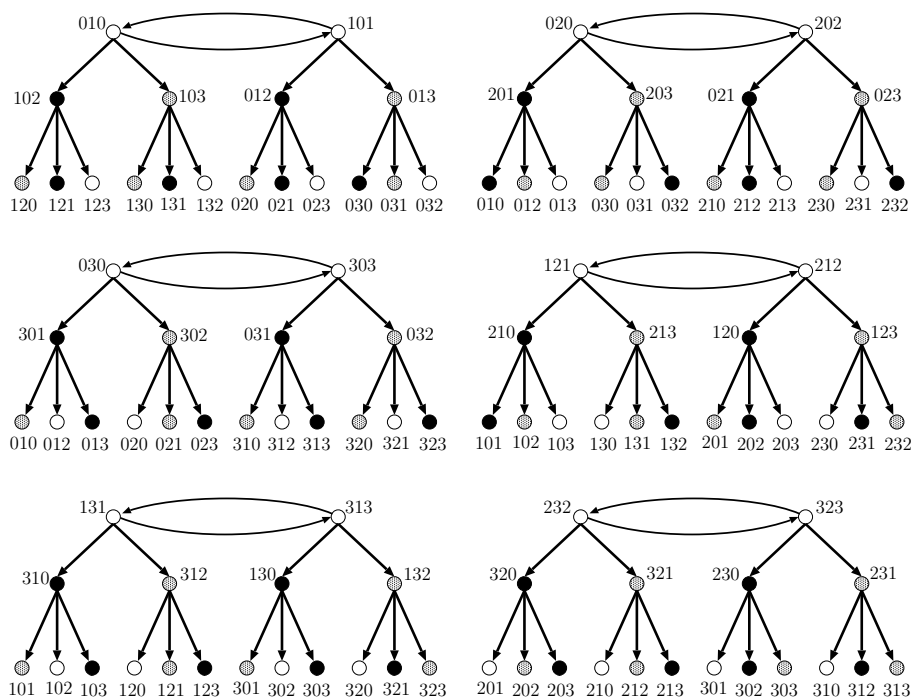


図 7.12: $K(2,3)$ の bijection によるラベリングの例

$0 \leq i, j \leq d-1, i \neq j$ とおく . このとき , $\forall v \in V(v_0, v_1, \dots, v_{n-2}), i \neq j \Rightarrow L_i(v) \neq L_j(v)$.

例として , ある bijection L_0 に対し $L_i(v) = (L_0(v) + i) \bmod d$ を定義すると , L_0, L_1, \dots, L_{d-1} は補題 7.31 の条件を満たす d 個の bijection である . $K(2,3)$ について , さらに別の bijection によりラベル付けしたものを図 7.12 に示す . ただし , 各 2-cycle-rooted tree 上において , 内側が白の頂点はその cycle-rooted tree 内でのラベルが 0 のものであり , 内側が黒のものはラベルが 1 のものを , 網掛けのものはラベルが 2 のものをそれぞれ表している .

これらの bijection を用いることで , phase 2. に必要なラウンド数が次で与えられる .

補題 7.32 phase 2. に必要なラウンド数は $dn - 1$ ラウンドである .

証明 phase 1. によって各 2-cycle 上の頂点に一つずつ情報が送られたラウンドを 0 ラウンド目として扱うことにする . phase 1. の操作から , $K(d,n)$ 中で 2 種類の文字の交互列をラベルとして持つ $d(d+1)$ 個の頂点は それぞれ異なる情報の一つずつ持っている . さらに , これらの頂点は $d(d+1)/2$ 個の 2-cycle-rooted tree のいずれか一つの上で cycle-vertex となる . phase 2. では各 2-cycle-rooted tree $\langle A_{v_0, v_1} \rangle$ 上において , 二つの cycle-vertex が持つ情報を同じ部分ダイグラフに属するすべての頂点へ送信する . これらの送信スキームを各 2-cycle-rooted tree で独立に与えることは可能であるが , 実際には $K(d,n)$ 上で同時に行なわれる操作であるため同一ラウンド中に一つの頂点が二つ以上の情報を送信および受信しないようにする必要がある . 各 2-cycle-rooted tree は完全 d 進であり , それぞれが弧素であるため , 各頂点はある一つの 2-cycle-rooted tree 上でしか送信は行なわず , 同一ラウンドに複

数の情報を送信するような場合は現れない．一方で，各頂点が同一ラウンドに複数の情報を受信しないようにするためには情報を持った頂点がどのような順番で隣接頂点に情報を送信していくかを定める必要がある．この送信順序を補題 7.31 により得られる d 個の bijection L_0, L_1, \dots, L_{d-1} からのラベルをもとにして決定することを考える． $V(v_0, v_1, \dots, v_{n-2})$ の各頂点に対し，それらが含まれる d 個の 2-cycle-rooted tree 上で，それぞれ異なる d 個の bijection L_0, L_1, \dots, L_{d-1} により写像されるラベルを新たに与える．ただし， $\langle A_{v_0, v_1} \rangle$ または $\langle A_{v_1, v_0} \rangle$ 内において， $L_{v_0}((v_0, v_1, v_0, v_1, \dots)) = 0, L_{v_1}((v_1, v_0, v_1, v_0, \dots)) = 0$ を満たすものを用いる．これらの割り当てをもとに，各部分ダイグラフ内で葉でない頂点 $(x_0, x_1, x_2, \dots, x_{n-1})$ は d 個の隣接頂点 $(x_1, x_2, \dots, x_{n-1}, \alpha)$ に情報を送る順番として， d を法として送信する際のラウンド数と等しいラベルが割り当てられている隣接先に情報を送るものとする．この場合，各頂点はそれが含まれる d 個の 2-cycle-rooted tree で異なるラウンドで情報を受信するため，phase 2. において複数の情報を同時に受信することはない．さらに，各 2-cycle 上の頂点はそれぞれ d ラウンド目にもう一方の cycle-vertex から情報を受信するため，一つ目の情報を送信したのと同様の順番で二つ目の情報を $d+1$ ラウンド目から送信開始できる．この情報は残り $d-1$ 個の子にのみ送信すればよいので，各 cycle-vertex が二つの情報をすべての隣接頂点に送信するために必要なラウンドは $2d-1$ ラウンドである．さらに，各部分ダイグラフ上で頂点が二つ目の情報を受信するのは一つ目の情報を受信してから d ラウンド後であるので，すぐに二つ目の情報の送信を始めることができる．cycle-vertex の隣接先の頂点で二つ目の情報を最後に受信するのは $d-1$ のラベルを持つ隣接頂点であり，各 2-cycle-rooted tree の高さは $n-1$ であるから，この部分グラフ内のすべての頂点が cycle-vertex が持つ二つの情報を受信するのに必要なラウンド数は $2d-1 + (n-2)d = dn-1$ となる． \square

phase 2. により，各 $\langle A_{v_0, v_1} \rangle$ に含まれるすべての頂点は cycle-vertex が持つ二つの情報を受信する．今， (i, v_1, \dots, v_{n-1}) で表される頂点は $\langle A_{i, x} \rangle$ あるいは $\langle A_{x, i} \rangle$, ($0 \leq x \leq d, x \neq i$) で表されるすべての 2-cycle-rooted tree に含まれているため，phase 2. で (i, x_1, \dots, x_{n-1}) が得る情報は (i, x, i, x, \dots) あるいは (x, i, x, i, \dots) で表される 2-cycle 上の頂点から送られた $2d$ 個の情報である．この関係を図 7.13 に表す．図 7.13 内の四角で囲まれた値は対応する頂点が phase 2. 終了時に持っている情報を表す．これをもとに phase 3. で残りの $d(d-1)$ 個の情報を送信することを考える．

補題 7.33 phase 3. に必要なラウンド数は $d(d-1)$ ラウンドである．

証明 phase 2. と同様に，各頂点は d を法として現在のラウンド数と等しいラベルが割り当てられている隣接先に情報を送るものとし，一つの情報を d 個の隣接頂点に送り終えてから次の情報を送信し始めるものとする．また，頂点 $(j, v_0, v_1, \dots, v_{n-2})$, ($0 \leq j \leq d, j \neq v_1$) は隣接頂点 $(v_0, v_1, \dots, v_{n-2}, \alpha)$ に $(j, \beta, j, \beta, \dots)$, ($0 \leq \beta \leq d, \beta \neq j, v_0$) で表される $d-1$ 個の cycle-vertex から送られ始めた情報の送信を行なう．このとき， $(j, v_0, v_1, \dots, v_{n-2})$ で表

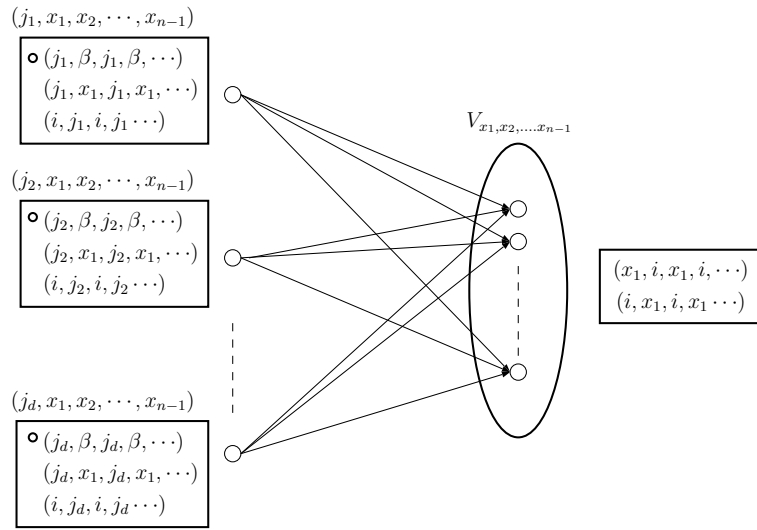


図 7.13: phase 2. 終了時に各頂点が持つ情報

される d 個の頂点は $(v_0, v_1, \dots, v_{n-2}, \alpha)$ で表される各頂点にそれぞれ異なる d 個の情報を送信する．各頂点が隣接先の d 個の頂点すべてに $d - 1$ 個の情報を送信するには $d(d - 1)$ ラウンド必要であるため，頂点 $(v_0, v_1, \dots, v_{n-2}, \alpha)$ は $d(d - 1)$ ラウンド後にすべての情報を受信したことになる．これは $K(d, n)$ 上のすべての頂点に対して成り立つため，命題は真となる． \square

特に phase 3. において各頂点はすべての頂点がすべての情報を受信するまで毎ラウンド情報を送受信していることがいえる．以上の結果から， $K(d, n)$ 上の高々 $d(d + 1)$ 個の情報をマルチソースブロードキャストを行うのに必要なラウンド数の上界として，次が成り立つ．

定理 7.34 $K(d, n)$ 上で $d(d + 1)$ 個の情報をマルチソースブロードキャストするために必要なラウンド数は高々 $d^2 + 2dn - d + n - 1$ ラウンドである．

7.3 各アルゴリズムに必要なラウンド数の比較

この節では本章で述べた各アルゴリズムを実行する際に必要なラウンド数の上界を比較する．[41] において任意のグラフに対するマルチソースブロードキャストに必要なラウンド数の下界が示されているが，これは各頂点が 1 ラウンドに情報の受信と送信のどちらかのみを行なえるモデルにおける結果である．グラフの頂点数を N ，情報の総数を M とおくと，この下界は $\lceil \log_2(N/M) \rceil + 2(M - 1)$ で表される． $\lceil \log_2(N/M) \rceil$ はグラフ上の各頂点がそれぞれ少なくとも一つの情報を受信するために必要なラウンドの下界であり，本研究で用いている同時送受信モデルにおいても同様の下界を持つ． $2(M - 1)$ は各頂点が残りの高々

$M - 1$ 個の情報をすべて受信するために必要なラウンド数であり，同時送受信モデルにおいてこれに必要なラウンド数は半分で済む．したがって同時送受信モデルにおけるマルチソースブロードキャストの下界は次で示される．

定理 7.35 ダイグラフ G の頂点数を N とおき， G が持つ情報の総数を M とおく．このとき，同時送受信モデルにおいてマルチソースブロードキャストを行なうには少なくとも $\lfloor \log_2(N/M) \rfloor + M - 1$ ラウンド必要である．

de Bruijn ダイグラフ $B(d, n)$ に対して，[32] で得られるブロードキャストを d 回行ったラウンド数を 1 としたときの，提案した各アルゴリズムおよび下界の比率が図 7.14 に示されている．図中において，My result は今回提案したアルゴリズムを，Known result は [37] で提案されたアルゴリズムをそれぞれ表している．この図から，提案したアルゴリズムはブロードキャストアルゴリズムを複数回行う方法のおよそ 3 割以下のラウンド数で実行可能であることがわかる．また，[37] で提案されたアルゴリズムは d が n の値より極端に大きくなる場合ではブロードキャストの手法に劣ってしまう可能性があるが，本アルゴリズムではそのような場合でも効率よく情報散布することが可能である．これは， d が n より十分大きな場合の各アルゴリズムのラウンド数が本アルゴリズムと下界では $O(d)$ ，ブロードキャストの応用が $O(d \log d)$ の一方で [37] の手法では $O(d^2)$ となってしまうためである．

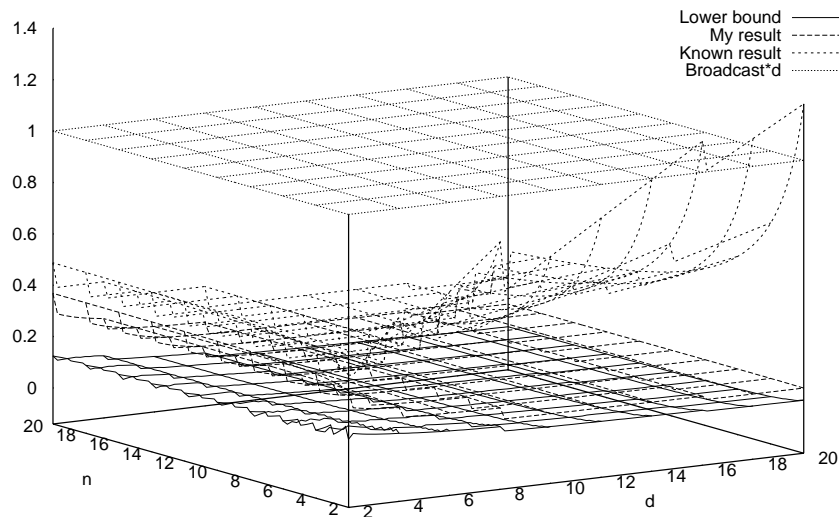


図 7.14: $B(d, n)$ における各アルゴリズムのラウンド数の比較

Kautz ダイグラフ $K(d, n)$ に対しても，[32] で得られるブロードキャストを $d(d+1)$ 回行ったラウンド数を 1 としたときの，提案した各アルゴリズムおよび下界の比率を図 7.15

に示す．図中において，2-cycle は 2-cycle-rooted tree による分解を用いたアルゴリズムを，Isomorphic Factorization は 2-cycle-rooted tree による同型因子分解を用いたアルゴリズムを，SPCRT は大きな root-cycle を持つ全域 cycle-rooted tree を用いたアルゴリズムをそれぞれ表している．ただし，2-cycle および SPCRT は一度の試行で同時に d 個までの情報を扱えるため，アルゴリズムを $d + 1$ 回繰り返して行っている． d が n より十分大きな場合には下界と 2-cycle, Isomorphic Factorization に必要なラウンド数は $O(d^2)$ であるのに対し，ブロードキャストの手法は $d^2 \log d$, SPCRT は d^3 となるため，SPCRT のアルゴリズムがブロードキャストの手法よりも劣る場合がある．

2-cycle, Isomorphic Factorization およびブロードキャストの手法に対して， $K(5, 5)$ 上で情報の総数が変化した場合の各アルゴリズムに必要なラウンド数の値を示す．ラウンド数を見てみると，情報の総数が少ない場合にはブロードキャストを数回繰り返すことがもっとも効率がよくなる．これは，提案した二つのアルゴリズムが同時に複数個の情報を扱うことを想定しているためである．したがって，2-cycle, Isomorphic Factorization の各アルゴリズムはそれぞれ情報の総数が任意の正整数 x に対し $(x - 1)d + 1$ から xd および $(x - 1)d(d + 1) + 1$ から $xd(d + 1)$ の間で必要なラウンド数が変化しない．一方で，2-cycle と Isomorphic Factorization のラウンド数を比較すると，情報の総数が d に近い場合には Isomorphic Factorization の方がラウンド数が少なくて済むが，総数が $d(d + 1)$ に近づくとつれて 2-cycle の方が効率がよくなることがわかる．

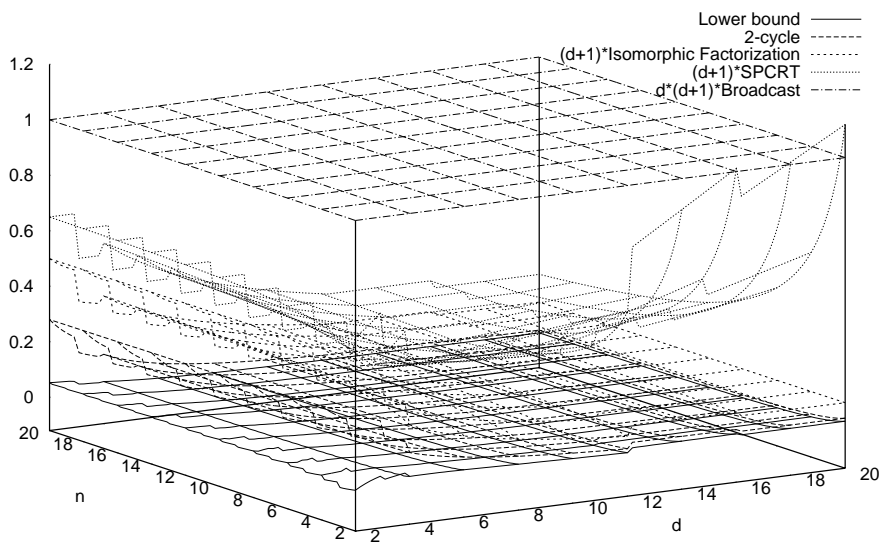


図 7.15: $K(d, n)$ における各アルゴリズムのラウンド数の比較

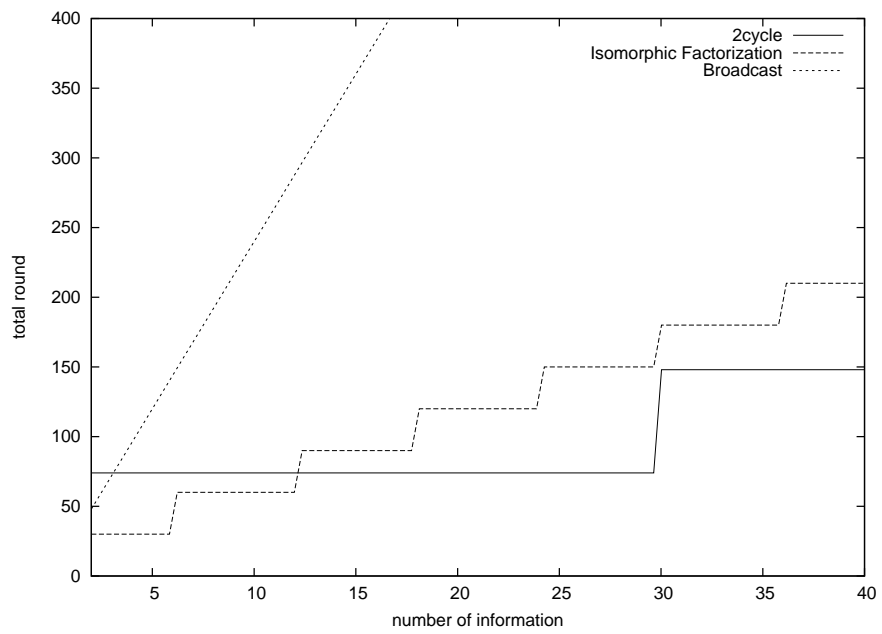


図 7.16: 情報の総数による各アルゴリズムのラウンド数の変化

第8章 まとめと今後の課題

本研究では相互結合網のモデルの一つである de Bruijn 族のグラフに対して、グラフの分解とグラフ上での情報散布問題という二つの観点から考察を行った。特に、cycle-rooted tree というグラフクラスとの関連性について着目することで一様性の高い分解を与え、その上で効率的な情報散布のアルゴリズムを提案した。本研究で得られた結果を以下にまとめる。

1. グラフの分解

- (a) de Bruijn, Kautz ダイグラフの同型な cycle-rooted tree による分解。

de Bruijn ダイグラフ $B(d, n)$ および Kautz ダイグラフ $K(d, n)$ に対して、 n 以下のサイクルが含む頂点の包含関係について定理 4.10 および定理 4.22 で示し、互いに素な root-cycle を持つ完全 d 進 cycle-rooted tree による分解を定理 4.11 および定理 4.23 で与えた。

- (b) 一般化 de Bruijn, 一般化 Kautz ダイグラフの cycle-rooted tree による分解。

一般化 de Bruijn ダイグラフ $G_B(d, n)$ および一般化 Kautz ダイグラフ $G_K(d, n)$ に対して loop-rooted tree による因子分解が可能であるための必要十分条件を与え、実際に因子分解の方法を定理 4.29 および定理 4.31 で与えた。また、上記の条件を満たさない場合に対してはループを複数含むようなダイグラフを含む同型因子分解を与えた。さらにいくつかの条件を満たすような d と n に対して $G_B(d, n)$ の loop-rooted tree による同型因子分解を定理 4.36 で与えた。

- (c) 準同型写像を用いた de Bruijn ダイグラフのサイクル分解

de Bruijn ダイグラフ $B(d, n)$ から $B(d, n-1)$ への準同型写像を再帰的に用いることで $B(d, n)$ から K_d^* への準同型写像を定義し、この写像をもとに分割される頂点集合がサイクルを導くことを定理 5.5 で示した。

- (d) 一般化 de Bruijn ダイグラフのクロネッカー積の同型因子分解

一般化 de Bruijn ダイグラフのクロネッカー積 $G_B(d, m) \otimes G_B(d, n)$ が持つループの総数 $L_m \otimes L_n$ が $G_B(d, mn)$ が持つループの総数 L_{mn} の間の整除性について考察を行い、クロネッカー積の一方が完全対称ダイグラフの場合には一般化 de Bruijn ダイグラフで同型因子分解可能であることを示した (定理 5.12)。

$$G_B(d, mn) \mid K_m^+ \otimes G_B(d, n).$$

2. グラフ上での情報散布

(a) de Bruijn ダイグラフ上でのマルチソースブロードキャストイング

de Bruijn ダイグラフ $B(d, n)$ 上で loop-rooted tree による同型因子分解を用いたマルチソースブロードキャストイングのアルゴリズムを与え, 必要なラウンド数の上界を求めた (7.1.2 小節) .

(b) Kautz ダイグラフ上でのマルチソースブロードキャストイング

Kautz ダイグラフ $K(d, n)$ に対し, cycle-rooted tree によるマルチソースブロードキャストイングアルゴリズムをいくつか提案し, 必要なラウンド数の上界を求めた .

i. 大きな root-cycle を持つ全域 cycle-rooted tree 上でのアルゴリズム

7.2.1 小節で $K(d, n)$ の 2-cycle-rooted tree による同型因子分解により得られる各因子間の隣接関係について考察し, 各 2-cycle-rooted tree の cycle-vertex を少なくとも一つずつ含むような全域 cycle-rooted tree の構成方法を与えた . また, その上で同時に d 個までの情報を扱うアルゴリズムを 7.2.2 小節で提案した .

ii. 2-cycle-rooted tree による同型因子分解を用いたアルゴリズム

$K(d, n)$ を d 個の 2-cycle-rooted tree により同型因子分解し, 各因子の上でブロードキャストイングを情報の遅延が発生しないように与えることで高々 d 個の情報を $K(d, n)$ 上でのマルチソースブロードキャストイングを行うアルゴリズムを 7.2.4 小節で提案した .

iii. 完全 d 進 2-cycle-rooted tree による分解を用いたアルゴリズム

$K(d, n)$ を $\binom{d+1}{2}$ 個の完全 d 進 2-cycle-rooted tree により分解し, $d(d+1)$ 個の cycle-vertex からそれぞれ情報散布を行うことで同時に $d(d+1)$ 個の情報を扱うマルチソースブロードキャストイングのアルゴリズムを 7.2.4 小節で提案した .

本研究の今後の課題として, de Bruijn 族グラフの分解に関する研究とマルチソースブロードキャストイングに関する研究の二つに分けて以下に述べる .

1. de Bruijn 族グラフの分解

de Bruijn ダイグラフ $B(d, n)$ および Kautz ダイグラフ $K(d, n)$ に対して, 長さ n 以下のサイクルによる分解を本研究で示した . 一方で, 長さが n より大きなサイクルについては各ダイグラフ上での数え上げの問題は未解決である . また, 本研究での分解では完全 d 進 cycle-rooted tree の深さは他のパラメータに依存し一意に与えられており, 深さをパラメータとした分解に関する研究が残されている .

一般化 de Bruijn ダイグラフ $G_B(d, n)$ および一般化 Kautz ダイグラフ $G_K(d, n)$ に対しては loop-rooted tree による因子分解可能であるための必要十分を与えたが, loop-rooted tree により同型因子分解が可能かどうかは $G_B(d, n)$ の特別な場合のみしか与えておらず, d と n に関する必要十分条件を与える課題が残っている.

また, cycle-rooted tree 以外による分解に対して, Kautz ダイグラフは de Bruijn ダイグラフと同様に準同型写像によるサイクル分解が可能であると予想している. さらに任意の一般化 de Bruijn ダイグラフのクロネッカー積 $G_B(d, n) \otimes G_B(d, m)$ に対しての一般化 de Bruijn ダイグラフによる分解は未解決のままである.

2. de Bruijn 族グラフ上でのマルチソースブロードキャスト

本研究では de Bruijn ダイグラフ $B(d, n)$ および Kautz ダイグラフ $K(d, n)$ 上でのマルチソースブロードキャストのアルゴリズムを提案したが, いずれも同時に扱える情報の総数には制限が存在する. この制限は分解する cycle-rooted tree の cycle-vertex の総数に強く依存しており, 本研究で得られた cycle-rooted tree による分解を用いることでより多くの情報を同時に扱うアルゴリズムを構成できると予想している. Kautz ダイグラフ上において高々 d 個の情報を同時に扱えるアルゴリズムと高々 $d(d+1)$ 個の情報を同時に扱えるアルゴリズムのラウンド数を比較したが, 情報の総数が d に近い値の場合には前者の方が効率がよいことが得られている. これにより, 残りの情報の総数に応じてこれらのアルゴリズムを使い分けることでより高速にマルチソースブロードキャストを行うアルゴリズムを構成できると考えている.

謝辞

本研究は学部4年，博士前期および後期課程を通じて行われたものです．本研究において，柴田 幸夫 名誉教授には多大な御指導，御鞭撻を賜りました．研究方針，研究過程においてだけでなく，生活面においても薫陶を受けたことに深く感謝致します．

また，群馬大学大学院工学研究科情報工学専攻 荒木 徹 准教授には一年間という短い期間にもかかわらず多くの御指導並びに有益な御助言を賜り感謝致します．本研究をこのような一つの形で纏め上げることができたのもひとえに両先生のお陰であり，重ねて感謝申し上げます．

貴重な時間を割いて本論文の副査をして頂き，有益なアドバイスをしてくださった群馬大学大学院工学研究科情報工学専攻 横尾 英俊 教授，山崎 浩一 教授，天野 一幸 准教授には深く感謝すると共に厚く御礼申し上げます．特に 横尾 英俊 教授 につきましてはリサーチプロポーザルの指導教員としても御助言頂きました．

さらに，研究過程において多くの御助言をくださった群馬大学大学院工学研究科情報工学専攻 大澤 新吾 助教，快適な環境で研究できるように他方面においてお世話になりました群馬大学工学部情報工学科 鎗木 喜雄 技術専門職員をはじめとする研究室の方々には心より感謝の意を表します．また，研究室の先輩でもある現 群馬大学大学院工学研究科生産システム工学専攻 田中 勇樹 助教には日頃の研究生生活においてお世話になり，ここに深く感謝致します．

最後に，家族をはじめとする私を支えてくださったすべての方々に対し感謝の意を表します．

参考文献

- [1] A. Bar-Noy and S. Kipnis, “Broadcasting multiple messages in simultaneous send/receive systems,” *Discrete Appl. Math.*, Vol. 55, pp. 95-105, 1992.
- [2] D. Barth, P. Fragopoulou and M.-C. Heydemann “Uniform emulations of Cartesian-product and Cayley graphs,” *Discrete Appl. Math.*, Vol. 116, No. 1-2, pp. 37-54, 2002.
- [3] J-C. Bermond and P. Fraigniaud, “Broadcasting and Gossiping in de Bruijn networks,” *SIAM J. Comput.*, Vol. 23, pp. 212-225, 1994.
- [4] J-C. Bermond and P. Hell, “On even factorizations and the chromatic index of the Kautz and de Bruijn digraphs,” *J. Graph Theory*, Vol. 17, No. 5, pp. 647-655, 1993.
- [5] J-C. Bermond and C. Peyrat, “de Bruijn and Kautz networks : a competitor for the hypercube?,” in F. André and J. P. Verjus (Eds.), *Hypercube and Distributed Computers*, North Holland, 1989, pp.279-293.
- [6] J.-C. Bermond and P. Hell, “On even factorizations and the chromatic index of the Kautz and de Bruijn digraphs,” *J. Graph Theory*, Vol.17, pp. 647-655, 1993.
- [7] J.-C. Bermond, P. Hell, A. L. Liestman and J. G. Peters, “Broadcasting in bounded degree graphs,” *SIAM J. Discrete Math.*, Vol. 5, No. 1, pp. 10-24, 1992.
- [8] H. L. Bodlaender, “The complexity of finding uniform emulations on paths and ring networks,” *Information and Computation*, Vol. 86, No.1, pp. 87-106, 1990
- [9] H. L. Bodlaender, J. van Leeuwen, “Simulation of large networks on smaller networks,” *Information and Control*, Vol. 71, No.3, pp. 143-180, 1986
- [10] J. Bruck, R. Cypher and C.T. Ho, “Multiple message broadcasting with generalized Fibonacci trees,” *Proceedings of the 4th IEEE Symposium on Parallel and Distributed Processing*, pp. 424-431, 1992.
- [11] G. Chartrand, L. Lesniak, *Graphs & Digraphs*, Fourth edition, Chapman & Hall/CRC, 2004.

- [12] G.-L. Chia and P.-H. Ong, "Isomorphic factorization of the complete graph into Cayley digraphs," *J. Graph Theory*, Vol.52, No. 3, pp. 243-256, 2006.
- [13] M.-L. Chia, D. Kuo and M.-F. Tung, "The multiple originator broadcasting problem in graphs," *Discrete. Appl. Math.*, Vol. 155, No. 10, pp. 1188-1199, 2007.
- [14] G.-M. Cui, "A fault-tolerant broadcasting algorithm for hypercubes," *Information Processing Letters*, Vol. 66, pp. 93-99, 1998.
- [15] K. Diks, S. Dobrev, E. Kranakis and P. Ruzicka, "Broadcasting in unlabeled hypercubes with linear number of messages," *Information Processing Letters*, Vol. 66, pp. 181-186, 1998.
- [16] M. Dietzfelbinger, "Gossiping and broadcasting versus computing functions in networks," *Discrete. Appl. Math.*, Vol. 137, pp. 127-153, 2004.
- [17] D. Z. Du and D. F. Hsu, "De Bruijn digraphs, Kautz digraphs, and their generalizations," in Ding-Zhu Du and D.Franc Hsu,eds., *Combinatorial Network Theory*, pp. 65-105, 1996.
- [18] D. Z. Du, D. F. Hsu, F. K. Hwang and X. M. Zhang, "The Hamiltonian property of generalized de Bruijn digraphs," *J. Combin. Theory, Series B*, Vol. 52, No. 1, pp. 1-8, 1991.
- [19] D. Z. Du and F. K. Hwang, "Generalization de Bruijn Digraphs," *NETWORKS*, Vol. 18, pp. 27-38, 1988.
- [20] A.-H. Esfahanian and S. L. Hakimi, "Fault-Tolerant Routing in DeBruijn Comrnunication Networks," *IEEE Transactions on Computers*, Vol. C-34, No. 9, pp. 777-788, 1985.
- [21] A. Farlry and A. Proukurowski, "Gossiping in grid graphs," *J. Combin. Inform. System Sci.*, Vol.5, pp. 161-172, 1980.
- [22] J. P. Fishburn, R. A. Finkel, "Quotient Networks," *IEEE Transactions on Computers*, Vol. 31, No. 4, pp. 288-285, 1982.
- [23] M. A. Fiol, J. L. A. Yebra, I. Alegre, "Line digraph iterations and the (d, k) digraph problem," *IEEE Transactions on Computers*, Vol. C-33, pp. 400-403, 1984.
- [24] P. Fraigniaud and E. Lazard, "Methods and problems of communication in usual networks," *Discrete. Appl. Math.*, Vol. 53, pp. 79-133, 1994.

- [25] R. Graham, D. E. Knuth, O. Patashnik, “Concrete Mathematics: A Foundation for Computer Science,” Addison-Wesley, 1994.
- [26] G. Hahn, M.-C. Heydemann, J. Opatrny, and D. Sottau, “Covering trees and broadcasting in de Bruijn graphs,” Research Report, Concordia University.
- [27] T. Hasunuma, “Completely independent spanning trees in the underlying graph of a line digraph,” *Discrete Math.*, Vol. 234, pp. 149-157, 2001.
- [28] T. Hasunuma, Y. Kikuchi, T. Mori, Y. Shibata, “On the number of cycles in generalized Kautz digraphs,” *Discrete Math.*, Vol. 285, pp. 127-140, 2004.
- [29] T. Hasunuma and Y. Shibata, “Counting small cycles in generalized de Bruijn graphs,” *NETWORKS*, Vol. 29, pp. 39-47, 1997.
- [30] T. Hasunuma, Y. Shibata, “Isomorphic decomposition and arc-disjoint spanning trees of Kautz digraphs,” *IPSJ SIG Notes*, Vol. 96-AL-51, pp. 63-70, 1996.
- [31] S. M. Hedetniemi and S. T. Hedetniemi, “A survey of gossiping and broadcasting in communication networks,” *NETWORKS*, Vol. 18, pp. 319-349, 1988.
- [32] M.-C. Heydemann, J. Opatrny, D. Sottau, “Broadcasting and spanning trees in de Bruijn and Kautz networks,” *Discrete. Appl. Math.*, Vol. 37/38, pp. 297-317, 1992.
- [33] J. Hromkovič, R. Klasing, B. Monien and R. Peine, “Dissemination of information in Interconnection Networks (Broadcasting & Gossiping),” in D.-Z. Du and D. F. Hsu (Eds.), *Combinatorial Network Theory*, Springer, pp.125-212, 1996.
- [34] M. Imase and M. Itoh, “Design to minimize diameter on building-block network,” *IEEE Trans. Comput.*, Vol. 30, no. 6, pp. 439-442, 1981.
- [35] M. Imase and M. Itoh, “A design for directed graphs with minimum diameter,” *IEEE Trans. Comput.*, Vol. 32, no. 8, pp. 782-784, 1983.
- [36] 入野 裕章, “de Bruijn ネットワークにおける cycle-rooted tree を用いた情報散布に関する研究,” 群馬大学大学院 工学研究科情報工学専攻 修士論文, 2006.
- [37] H. Irino, Y. Tanaka, H. Kawai, S. Osawa and Y. Shibata, “Broadcasting multiple messages by cycle-rooted tree,” *IPSJ SIG Technical Report*, 2007-AL-112(6), May 2007.
- [38] S. Johnson and C.-H. Ho, “Optimum broadcasting and personalized communication in hypercubes,” *IEEE Trans. Comp.*, Vol. C-38, pp. 1249-1268, 1989.

- [39] H. Kawai, N. Fujikake and Y. Shibata, "Factorization of de Bruijn digraphs by cycle-rooted trees," *Information Processing Letters*, Vol. 77, pp. 269-275, 2001.
- [40] H. Kawai, Y. Shibata, "Isomorphic factorization of the Kronecker product of generalized de Bruijn digraphs," *RIMS Kokyuroku*, Vol.1205, pp. 200-205, 2001.
- [41] S. Khuller, Y.-A. Kim, Y.-C. Wan, "On generalized gossiping and broadcasting," *J. Algorithms*, Vol.59, pp. 81-106, 2006.
- [42] N. S. Mendelsohn, "Directed graphs with the unique path property," in P. Erdos, A. Renyi and V. T. Sos (Eds.), *Combinatorial Theory and its Applications*, North Holland, 1970, pp.783-799.
- [43] 水野 雅之, "de Bruijn **ダイグラフ**の 2-cycle-rooted tree による分解," 群馬大学工学部 情報工学科 卒業論文, 2009.
- [44] S. Perennes, "Broadcasting and gossiping on de Bruijn, shuffle-exchange and similar networks," *Discrete. Appl. Math.*, Vol. 83, pp. 247-262, 1998.
- [45] D. K. Pradhan, "Fault-tolerant multiprocessor and VLSI-based systems communication architecture," *Fault Tolerant Computing Theory and Techniques*, Vol. II, Prentice-Hall, Englewood Cliffs, NJ, 1986.
- [46] S. M. Reddy, D. K. Pradhan and J. G. Kuhl, "Directed graphs with minimal diameter and maximal connectivity," *School of Engineering Oakland Univ. Tech. Rep.*, 1980.
- [47] R. A. Rowley and B. Bose, "Fault-Tolerant Ring Embedding in de Bruijn Networks," *IEEE Transactions on Computers*, Vol. 42, No. 12, pp. 1480-1486, 1993.
- [48] J.-L. Ruiz and M. Mora, "De Bruijn and Kautz digraphs of a rooted tree," *Discrete. Math.* Vol.293, No. 1-3, pp. 219-236, 2005.
- [49] G. Salamon, G. Wiener, "On finding spanning trees with few leaves" *Information Processing Letters*, Vol.105, No. 5, pp. 164-169, 2008.
- [50] Y. Shibata, Y. Gonda, "Extension of de Bruijn digraph and Kautz digraph," *Computers Math. Applic.*, Vol.30, No.9, pp. 51-61, 1995.
- [51] Y. Shibata, T. Hasunuma, and S. Fukada, "Isomorphic factorization of de Bruijn digraphs," *Discrete. Math.* Vol.218, pp. 199-208, 2000.
- [52] Y. Shibata, M. Shirahata and S. Osawa, "Counting closed walk in generalized de Bruijn graphs," *Information Processing Letters*, Vol.49, pp. 135-138, 1994.

- [53] P. J. Slater, E. Cookayne and S. T. Hedetniemi, "Information dissemination in trees," SIAM J. Comput., Vol. 10, pp. 692-701, 1981.
- [54] T. Tanaka, H. Kawai and Y. Shibata, "Isomorphic factorization, the Kronecker product and the line digraph," Information Processing Letters, Vol. 101, pp. 72-77, 2007.
- [55] 津野崇寛, 柴田幸夫, "Kautz ダイグラフ上の cycle-rooted tree を用いたマルチソースブロードキャストイング," 電子情報通信学会論文誌 A, Vol. J91-A, No. 2, pp. 212-222, 2008.
- [56] T. Tsuno, Y. Shibata, "Factorization of generalized de Bruijn and Kautz digraphs by loop-rooted trees," The 7th Japan Conference on Computational Geometry and Graphs, 2009.
- [57] T. Tsuno, Y. Shibata, "Multisource broadcasting on de Bruijn and Kautz digraphs using isomorphic factorizations into cycle-rooted trees," IEICE Trans. Fundamentals, Vol. E92-A, no. 8, pp. 1757-1763, 2009.
- [58] T. Tsuno, Y. Shibata, "An algorithm for multi-source broadcasting on Kautz digraphs using 2-cycle rooted trees," IEICE Trans. Fundamentals, Vol. E93-A, no. 10, pp. 1800-1805, 2010.
- [59] P. Tvrdík, "Kautz Necklaces," Research Report 94-08, LIP ENSL, 69364 Lyon, France, March, 1994.
- [60] P. Tvrdík, R. Harbane, M.-C. Heydemann, "Uniform homomorphism of de Bruijn and Kautz networks," Discrete Appl. Math., Vol. 83, pp. 279-301, 1998.
- [61] M. Walsh, "Minimum broadcast tree decompositions" Discrete. Math. Vol.309, pp. 2620-2625, 2009.
- [62] I. Wojciechowska, "Broadcasting in grid graphs," Ph.D. dissertation, West Virginia University, 1999.
- [63] W. S. Wojciechowski and A. S. Wojcik, "Automated Design of Multiple-Valued Logic Circuits by Automatic Theorem Proving Techniques," IEEE Transactions on Computers, Vol. C-32, No. 9, pp. 785-798, 1983.
- [64] M. Yoeli, "Binary ring sequences," American Math. Monthly, Vol. 69, pp. 852-855, 1962.

付録 A 本論文で与えた定理の別証明

ここでは、本文中で証明を行った定理についての別の証明方法を記載する。4章では $B(d, n)$ および $K(d, n)$ に対し n 以下の任意の長さを持つ完全 d 進 cycle-rooted tree による分解が与えられることを定理 4.11 および定理 4.23 で証明した。

本文中では各有向サイクル間の距離の関係について言及し、それぞれを root-cycle とするような完全 d 進 cycle-rooted tree が互いに辺素であることを示した。以下では $B(d, n)$ のサイクル分解を与え、ダイグラフの growth を用いることで別解を与える。

$B(d, n)$ 上の長さ i のすべてのサイクルの集合を C^i と定義する。 $B(d, n)$ のサイクルによる分解について以下を示す。

定理 A.1 d, n を 1 以上の正整数とする。このとき、

$$B(d, n) = \bigoplus_{i \in I_{n+1}} C^i$$

が成り立つ。

証明 $V(B(d, n+1))$ から $A(B(d, n))$ への写像 ϕ を次のように定義する。

$$\phi : (u_0, u_1, \dots, u_n) \rightarrow ((u_0, u_1, \dots, u_{n-1}), (u_1, u_2, \dots, u_n))$$

このとき、 ϕ が bijection であることを示す。 $B(d, n+1)$ の任意の頂点 $u = (u_0, u_1, \dots, u_n)$ に対し、 $\phi(u) = ((u_0, u_1, \dots, u_{n-1}), (u_1, u_2, \dots, u_n))$ を満たすような $B(d, n)$ 上の弧は定義より必ず存在する。したがって ϕ は surjection である。一方で、 $B(d, n+1)$ 上での異なる二つの頂点 $u = (u_0, u_1, \dots, u_n), v = (v_0, v_1, \dots, v_n)$ に対して、 $\phi(u)$ と $\phi(v)$ が $B(d, n)$ 上の同一の弧 $((w_0, w_1, \dots, w_{n-1}), (w_1, w_2, \dots, w_n))$ に写像されると仮定すると、 $(u_0, u_1, \dots, u_{n-1}) = (v_0, v_1, \dots, v_{n-1}), (u_1, \dots, u_{n-1}, u_n) = (v_1, \dots, v_{n-1}, v_n)$ となり $u = v$ が成り立ち、 u と v は $B(d, n+1)$ 上で同一の頂点となり、 ϕ は injection である。以上より ϕ は bijection である。

k_1, k_2 を $n+1$ の約数であるとする。定理 4.5 より、 $B(d, n+1)$ において長さ k_1 の有向サイクルと長さ k_2 の有向サイクルは点素である。さらに定理 4.10 より、これらの有向サイクルに含まれる頂点の総数は d^{n+1} であり、この値は $|V(B(d, n+1))|$ と等しい。したがって、 $B(d, n+1)$ 上の各頂点 $u = (u_0, u_1, \dots, u_n)$ は長さが $k, k \in I_{n+1}$ であるような有向サイクルのただ一つにのみ含まれていることがいえる。そのような有向サイクル C 上で u が隣接する頂点を $u' = (u_1, u_2, \dots, u_{n+1})$ とおく。今、 $B(d, n)$ 上の弧について考えると、弧

$\phi(u) = ((u_0, u_1, \dots, u_{n-1}), (u_1, u_2, \dots, u_n))$ は弧 $\phi(u') = ((u_1, u_2, \dots, u_n), (u_2, u_3, \dots, u_{n+1}))$ へ隣接しており, C の各頂点を写像して得られる $B(d, n)$ 上の k 本の弧は長さ k の有向サイクルを誘導する. よって ϕ は $B(d, n+1)$ 上の長さ k ($k \in I_{n+1}$) の有向サイクルの頂点集合を $B(d, n)$ 上の長さ k の有向サイクルの弧集合へと写像する. $|V(B(d, n+1))| = |A(B(d, n))| = d^{n+1}$ であり, ϕ は bijection であるから, これらの有向サイクルの弧集合は $B(d, n)$ の分解である. 以上より命題が成り立つ. \square

定理 A.1 より, $B(d, n)$ は長さが $n+1$ の約数であるようなすべての有向サイクルによって分解されることがいえる. この有向サイクルによる分解をもとに, 4章で導入した coloring を用いることで定理 4.11 に対する別解を与える.

定理 4.11 h を n 以下の正整数とする. このとき,

$$B(d, n) = \bigoplus_{i \in I_h} \text{CRT}^{i, n+1-h}$$

が成り立つ.

証明 定理 A.1 より, $B(d, h-1)$ において $B(d, h-1) = \bigoplus_{i \in I_h} C^i$ が成り立つ. 今, 定義 4.34 の d -regular growth を $\bigoplus_{i \in I_h} C^i$ の各有向サイクルに対して再帰的に $n+1-h$ 回適用すると, 定理 4.35 より,

$$L^{n+1-h}(B(d, h-1)) = \bigoplus_{i \in I_h} (L \cdot \rho_d)^{n+1-h}(C^i),$$

が成り立つ. ここで, $(L \cdot \rho_d)^{n+1-h}(C^i)$ は (C^i) に含まれる各有向サイクルに対して d -regular growth を再帰的に $n+1-h$ 回適用して得られるダイグラフである. $L^{n+1-h}(B(d, h-1)) \cong B(d, n)$ であり, 高さ l の完全 d 進 cycle-rooted tree の d -regular growth により高さ $l+1$ の完全 d 進 cycle-rooted tree が得られる. したがって, $\bigoplus_{i \in I_h} (L \cdot \rho_d)^{n+1-h}(C^i) = \bigoplus_{i \in I_h} \text{CRT}^{i, n+1-h}$ であり, $B(d, n)$ は $\bigoplus_{i \in I_h} \text{CRT}^{i, n+1-h}$ により分解可能である. \square

de Bruijn ダイグラフ $B(d, n)$ の結果と同様に, Kautz ダイグラフ $K(d, n)$ に対しても定理 4.23 の別解が与えられる.

定理 A.2 d, n を 1 以上の正整数とする. このとき,

$$K(d, n) = \bigoplus_{i \in I_{n+1}^-} C^i \bigoplus_{j \in I_n^-} C^j$$

が成り立つ.

証明 $V(K(d, n+1))$ から $A(K(d, n))$ への写像 ψ を次のように定義する .

$$\psi : (u_0, u_1, \dots, u_n) \rightarrow ((u_0, u_1, \dots, u_{n-1}), (u_1, u_2, \dots, u_n))$$

このとき , ψ が bijection であることを示す . $K(d, n+1)$ の任意の頂点 $u = (u_0, u_1, \dots, u_n)$ に対し , $u_i \neq u_{i+1}$, $0 \leq i \leq n-1$ であるから , $K(d, n)$ 上に頂点 $(u_0, u_1, \dots, u_{n-1})$ および (u_1, u_2, \dots, u_n) が存在する . $K(d, n)$ の隣接条件により , $\psi(u) = ((u_0, u_1, \dots, u_{n-1}), (u_1, u_2, \dots, u_n))$ を満たすような弧が $K(d, n)$ に必ず存在する . したがって ψ は surjection である . 一方で , $K(d, n+1)$ 上での異なる二つの頂点 $u = (u_0, u_1, \dots, u_n)$, $v = (v_0, v_1, \dots, v_n)$ に対して , $\psi(u)$ と $\psi(v)$ が $K(d, n)$ 上の同一の弧 $((w_0, w_1, \dots, w_{n-1}), (w_1, w_2, \dots, w_n))$ に写像されると仮定すると , $(u_0, u_1, \dots, u_{n-1}) = (v_0, v_1, \dots, v_{n-1})$, $(u_1, \dots, u_{n-1}, u_n) = (v_1, \dots, v_{n-1}, v_n)$ となり $u = v$ が成り立ち , u と v は $K(d, n+1)$ 上で同一の頂点となるため , ψ は injection である . 以上より ψ は bijection である .

l_1, l_2 を $l_1, l_2 \in (I_{n+1}^- \cup I_n^-)$ を満たす整数とする . 定理 4.19 より , $K(d, n+1)$ において長さ l_1 の有向サイクルと長さ l_2 の有向サイクルは点素である . さらに定理 4.22 より , これらの有向サイクルに含まれる頂点の総数は $d^{n+1} + d^n$ であり , この値は $|V(K(d, n+1))|$ と等しい . したがって , $K(d, n+1)$ 上の各頂点 $u = (u_0, u_1, \dots, u_n)$ は長さが l , $(l \in (I_{n+1}^- \cup I_n^-))$ である有向サイクルのただ一つにのみ含まれていることがいえる . そのような有向サイクル C 上で u が隣接する頂点を $u' = (u_1, u_2, \dots, u_{n+1})$ とおく . 今 , $K(d, n)$ 上の弧について考えると , 弧 $\psi(u) = ((u_0, u_1, \dots, u_{n-1}), (u_1, u_2, \dots, u_n))$ は弧 $\psi(u') = ((u_1, u_2, \dots, u_n), (u_2, u_3, \dots, u_{n+1}))$ へ隣接しており , C の各頂点を写像して得られる $K(d, n)$ 上の k 本の弧は長さ k の有向サイクルを誘導する . よって ψ は $K(d, n+1)$ 上の長さ l , $(l \in (I_{n+1}^- \cup I_n^-))$ の有向サイクルの頂点集合を $K(d, n)$ 上の長さ l の有向サイクルの弧集合へと写像する . $|V(K(d, n+1))| = |A(K(d, n))| = d^{n+1} + d^n$ であり , ψ は bijection であるから , これらの有向サイクルの弧集合は $K(d, n)$ の分解である . 以上より命題が成り立つ . \square

$B(d, n)$ の場合と同様に , d -regular growth により , 定理 4.23 は次のように示せる .

定理 4.23 h を n 以下の正整数とする . このとき ,

$$K(d, n) = \bigoplus_{i \in I_h^-} \text{CRT}^{i, n+1-h} \bigoplus_{j \in I_{h-1}^-} \text{CRT}^{j, n+1-h}$$

が成り立つ .

証明 定理 A.2 より , $K(d, h-1)$ において $K(d, h-1) = \bigoplus_{i \in I_h^-} C^i \bigoplus_{j \in I_{h-1}^-} C^j$ が成り立つ . d -regular growth を $\bigoplus_{i \in I_h^-} C^i \bigoplus_{j \in I_{h-1}^-} C^j$ の各有向サイクルに対して再帰的に $n+1-h$

回適用すると，定理 4.35 より，

$$L^{n+1-h}(K(d, h-1)) = \bigoplus_{i \in I_h^-} (L \cdot \rho_d)^{n+1-h}(\mathbf{C}^i) \bigoplus_{j \in I_{h-1}^-} (L \cdot \rho_d)^{n+1-h}(\mathbf{C}^j),$$

が成り立つ． $L^{n+1-h}(K(d, h-1)) \cong K(d, n)$ であり，高さ l の完全 d 進 cycle-rooted tree の d -regular growth により高さ $l+1$ の完全 d 進 cycle-rooted tree が得られる．したがって， $\bigoplus_{i \in I_h^-} (L \cdot \rho_d)^{n+1-h}(\mathbf{C}^i) \bigoplus_{j \in I_{h-1}^-} (L \cdot \rho_d)^{n+1-h}(\mathbf{C}^j) = \bigoplus_{i \in I_h^-} \mathbf{CRT}^{i, n+1-h} \bigoplus_{j \in I_{h-1}^-} \mathbf{CRT}^{j, n+1-h}$ であり， $K(d, n)$ は $\bigoplus_{i \in I_h^-} \mathbf{CRT}^{i, n+1-h} \bigoplus_{j \in I_{h-1}^-} \mathbf{CRT}^{j, n+1-h}$ により分解可能である． \square