

グラフ上の演算について

—与えられたグラフから新しいグラフを構成する—

守屋 悦朗

概要

与えられたグラフから新しいグラフを定義するために役立つ様々な新しい演算を導入し、その効用について考察する。それらは、コピーを作る演算、グラフの一部に別のグラフを代入する演算、グラフとグラフを加減乗除する演算、グラフとグラフを頂点や辺で貼り合わせる演算、逆に、グラフをいくつかの成分に分解する演算などであり、それらのうちのあるものは既存の演算の一般化であり、あるものは新たに導入された演算である。これらの演算はグラフ理論の授業や著書の中で用いた結果、それらを使うことの効果・有用性が確かめられた。

New operations on graphs are introduced to describe new graphs from given graphs, and the effect of using the operations are examined by teaching in introductory classes of Graph Theory etc.

1 なぜ新しいグラフ演算が必要なのか

グラフ理論の入門授業などにおいて必要とされることの一つは、いろいろなグラフの例を示すことである。そのために、与えられたグラフから新しいグラフを定義する様々な方法が導入され用いられている。それにもかかわらず、著者は早稲田大学教育学部数学科におけるグラフ理論の授業においてグラフについて初めて学ぶ学生に教える際に、これこれこういったグラフを定義したいと思うのに、既存の演算だけではそれを式で表すことができないことにしばしば出くわした。そのため必然的に必要となったのが、思うままにグラフを定義できるような新しい演算を導入することであった。そういった演算は、実際に授業 [2] や著書 [3] の中で例や演習問題を提示する際に使ってみたり、受講者にいろんなグラフをいろんな演算を用いて表してもらうことをしたりした結果、その効果が確かめられたので、それをまとめたのが本論文である。

そもそも、グラフに関する用語は文献（や分野、特に、数学系か工学系か）によって微妙な違いがあり、初心者に教える際のネックの 1 つとなっている。それらは

- 多重辺 (multi-edge, parallel edge) を許すか否か
- 自己ループ (loop, self-loop) を許すか否か
- 道の種類（頂点あるいは辺が重複しない道の呼称 (path, walk, trail), 閉じた道の呼称 (circuit, cycle)）に関する用語

等であり、同じ用語でも文献によって違うものを表すことがしばしばであるが、本論文では基本として以下の定義を用いる [6, 7, 8, 17, 18, 19, 24, 23]。この場合、多重辺や自己ループを許すも

のを多重グラフ (多辺グラフ, multigraph) とか擬似グラフ (pseudograph) と呼ぶことが一般的である。

X を有限集合とすると、 $|X|$ で元の個数を表し、正整数 n に対し $\binom{X}{n}$ は $\{Y \mid Y \subseteq X, |Y| = n\}$ を表す。空でない有限集合 V と $E \subseteq \binom{V}{2}$ の対 $G = (V, E)$ を (無向) グラフ (undirected) graph) という。 G の頂点集合 V を $V(G)$ で、辺集合 E を $E(G)$ でも表す。 V の元を頂点 (vertex, point), E の元を辺 (edge, line) といい、 $|V|$ を G の位数 (order), $|E|$ を G のサイズ (size) という。 辺 $\{u, v\}$ は uv あるいは vu と略記する。

多重辺や自己ループも許すものをグラフと定義することも多く [5, 9, 10, 11, 14, 15, 16, 22], この場合、多重辺を許さないグラフを線形グラフ (linear graph) といい、多重辺も自己ループも許さないグラフを単純グラフ (simple graph) ということが多い。

また、道の呼称も書物や論文によって違いがあるものの1つである。文献 [3, 7, 16, 20, 19] では (i) 隣接している頂点の有限列 (辺や頂点の重複を許す) v_1, v_2, \dots, v_n ($v_i v_{i+1} \in E, 1 \leq i \leq n-1$) を道と呼んでいるが、[3, 19] では、(ii) すべての辺 $v_i v_{i+1}$ が異なるものを単純道 (simple path) といい、(iii) すべての頂点異なるものを基本道 (elementary path) と呼んでおり、[7, 21] をはじめとするアルゴリズム分野では後者 (iii) を単純道 (simple path) と呼ぶことが多いが、最も広く使われている呼称は (i), (ii), (iii) の順に歩道 (walk), 小径 (trail), 道 (path) である。

特に断りのない限り、本論文で用いる用語や記法は基本的に [8] に従う。

2 既存のグラフ演算とその拡張

まず、文献 [3]~[25] のいずれかで使われているグラフ演算を以下にまとめておくとともに、あるものについてはそれを一般化した演算を導入する。以下、 $G = (V, E)$ を任意のグラフとする。

(1) 辺と頂点の削除と追加

G からその1つの辺 $e \in E$ を除去して得られるグラフを $G - e$ で表す。すなわち、 $G - e := (V, E - \{e\})$ 。また、 G からその1つの頂点 $v \in V$ と v に接続するすべての辺を除去して得られるグラフを $G - v$ で表す。すなわち、 $G - v := (V - \{v\}, E - \{uv \mid u \in V\})$ 。

u, v が G の隣接しない2頂点のとき、 G に辺 uv を付け加えたグラフを $G + uv$ で表す。また、 G に新しい頂点 w を付け加え、 w と G のすべての頂点を結ぶ辺も付け加えたグラフを $G + w$ で表す。すなわち、 $G + uv := (V, E \cup \{uv\})$, $G + w := (V \cup \{w\}, E \cup \{vw \mid v \in V\})$ 。

さらに一般に、 $U = \{u_1, \dots, u_k\} \subseteq V$, $F = \{f_1, \dots, f_l\} \subseteq E$ に対して、

$$G - U := (\dots((G - u_1) - u_2) \cdots - u_k), \quad (2.1)$$

$$G - F := (\dots((G - f_1) - f_2) \cdots - f_l) \quad (2.2)$$

と定義する。これらを左辺のように書くのは、 u_1, \dots, u_k や f_1, \dots, f_l を削除する順番によらないためである。

同様に、 $U \cap V = \emptyset$ である $U = \{u_1, \dots, u_k\}$ および $F \cap E = \emptyset$ である $F = \{f_1, \dots, f_l\} \subseteq \binom{V}{2}$ に対して、グラフ $G + U$, $G + F$ を

$$G + U := (V \cup U, E \cup \{uv \mid u \in U, v \in V\}) \quad (2.3)$$

$$G + F := (V, E \cup F) \quad (2.4)$$

と定義する. $G + F \cong (\cdots((G + f_1) + f_2) \cdots + f_l)$ であるが, $G + U \not\cong (\cdots((G + u_1) + u_2) \cdots + u_k)$ である. $G + U$ と $G + F$ は [3, 5] 以外では使われていないが, 後述する, 2つのグラフの和の特別の場合 (U は (U, \emptyset) の省略形, F は (V, F) の省略形) と考えることもできる.

(2) 補グラフ

グラフ $G = (V, E)$ の辺の有無を入れ替えたグラフ, すなわち V を頂点集合とし

$$\bar{E} := \{uv \mid uv \in (\binom{V}{2} - E)\}$$

を辺集合とするグラフ \bar{G} を G の補グラフ (complement) という. この演算は次のように補グラフ化を G の部分グラフ G' に制限したものに拡張することができる: $G' = (V', E')$ を G の部分グラフとするとき,

$$\bar{G}|_{G'} := (V, (E - E') \cup \bar{E}').$$

(3) 誘導部分グラフ

$U \subseteq V(G)$ とする. U を頂点集合とし $\{uv \mid u, v \in U \text{ かつ } uv \in E(G)\}$ を辺集合とするグラフを, U から生成される G の点誘導部分グラフといい, $\langle U \rangle_G$ で表す. また, $F \subseteq E(G)$ のとき, $\{u, v \mid uv \in F\}$ を頂点集合とし F を辺集合とするグラフを, F から生成される G の辺誘導部分グラフ (induced subgraph) といい, $\langle F \rangle_G$ で表す. 誘導部分グラフを表す記法としては $\langle U \rangle_G, \langle F \rangle_G$ [3, 8, 14, 17, 24] あるいは $G[U], G[F]$ [5, 6, 23, 18, 11] が広く用いられているが, 後者は後述の合成演算と同じ記法なので, 合成演算も使う場合には誘導部分グラフには前者の記法が使われる.

(4) 和・差・共通部分

G_1 と G_2 の和 (union) を

$$G_1 \cup G_2 := (V_1 \cup V_2, E_1 \cup E_2) \quad (2.5)$$

によって, 差 (difference) を $G_1 - G_2 := (V_1 - V_2, E_1 - E_2 - \{uv \mid u \in V_2 \text{ または } v \in V_2\})$ によって定義する. また,

$$G_1 \cap G_2 := (V_1 \cap V_2, E_1 \cap E_2) \quad (2.6)$$

を G_1 と G_2 の共通部分 (intersection) という. 特に, $V := V_1 = V_2$ かつ $E_1 \cap E_2 = \emptyset$ であるとき,

$$G_1 \oplus G_2 := (V, E_1 \cup E_2) \quad (2.7)$$

と定義し ([5] では $G_1 \sqcup G_2$ と表している), 辺和 (edge sum) という [3, 8, 11, 17]. これらは頂点集合・辺集合の和・差・共通部分に倣って自然に定義されるものであるが, 記号 \oplus は集合の場合, 対称差

$X \oplus Y = (X \cup Y) - (X \cap Y)$ を表すのにもよく用いられるので, $G_1 \oplus G_2 := (G_1 \cup G_2) - (G_1 \cap G_2)$ と定義している例 [23] もある. 辺和は G_1 と G_2 の和としてよりも, $G_1 \oplus G_2$ を G_1 と G_2 に分解 (分割) する演算と見ることが多い.

和 $G_1 \cup G_2$ よりも ‘グラフらしい和’ としては結び (join)

$$V_1 \cap V_2 = \emptyset \text{ のとき, } G_1 + G_2 := (V_1 \cup V_2, E_1 \cup E_2 \cup \{v_1 v_2 \mid v_1 \in V_1, v_2 \in V_2\})$$

が広く用いられている [3, 8, 17, 24]. ただし, [11] では $V_1 \cap V_2 = \emptyset$ のときの $G_1 \cup G_2$ を表すのに $G_1 + G_2$ を用いている. しかし, 同じことは [11] よりも次に述べる $G_1 \boxplus G_2$ の方が一般的で使いやすくすぐれている.

(5) n 倍と素和

グラフ G の n 個のコピーを集めたグラフを nG で表すことは [8, 11, 17] で用いられているが, 数学的な定義としては曖昧である. そこで, [3] では G_1 と G_2 の素和 (disjoint union) を

$$G_1 \boxplus G_2 := (V_1 \cup \{u'_2 \mid u_2 \in V_2\}, E_1 \cup \{u'_2 v'_2 \mid u_2 v_2 \in E_2\})$$

と定義した (u'_2, v'_2 は $u'_2, v'_2 \notin V_1 \cup V_2$ なる新しい頂点). この定義では, G_1 や G_2 と共通部分のない, G_2 のコピー $G'_2 := (\{u'_2 \mid u_2 \in V_2\}, \{u'_2 v'_2 \mid u_2 v_2 \in E_2\})$ を作り, $G_1 \cup G'_2$ を $G_1 \boxplus G_2$

と定義している. この演算を使うと n 倍は $1G := G$, $nG := \overbrace{G \boxplus \cdots \boxplus G}^n$ ($n \geq 2$) と自然に定義でき, 非常に有用である.

次の命題は明らかである.

命題 2.1. 任意の n, m, m_1, \dots, m_k と任意の G, G_1, \dots, G_k に対して, 次が成り立つ.

- (1) $n(mG) \cong (nm)G$.
- (2) $n(m_1 G_1 \boxplus \cdots \boxplus m_k G_k) \cong nm_1 G_1 \boxplus \cdots \boxplus nm_k G_k$.

(6) 直積, n 乗, 閉包

G_1 と G_2 の直積 (Cartesian product)

$$G_1 \times G_2 := (V_1 \times V_2, \{(u_1, u_2)(v_1, v_2) \mid (u_1 v_1 \in E_1 \wedge u_2 = v_2) \vee (u_2 v_2 \in E_2 \wedge u_1 = v_1)\}) \quad (2.8)$$

は広く用いられている演算である [3, 8, 11, 14, 17, 24, 23]. これを ‘積’ と考えるなら, n 乗 (n -th power of G)

$$G^n := \begin{cases} K_1 & (n = 0) \\ \overbrace{G \times \cdots \times G}^n & (n \geq 1) \end{cases} \quad (2.9)$$

が自然に定義され (K_1 が単位元の役割を果たす), よく知られた n 次元超立方体 (n -cube) Q_n は $Q_n = K_2^n$ と表すことができる. しかし, n 乗のこのような定義は [3] で初めて導入されたものであり, これまで使われてきた n 乗は次のように定義するものであった [8, 14, 17]. $G = (V, E)$ を連結グラフとすると,

$$G^n := (V, \{uv \mid 1 \leq d(u, v) \leq n\}). \quad (2.10)$$

ここで, $d(u, v)$ は u, v 間の距離である. これと類似の演算に閉包 (closure) がある. G の位数を p とするとき, 演算 c と c^* を次のように定義する:

$$\begin{aligned} c(G) &:= (V, E \cup \{uv \mid uv \notin E, \deg(u) + \deg(v) \geq p\}), \\ c^n(G) &:= \begin{cases} G & (n = 0), \\ c(c^{n-1}(G)) & (n \geq 1), \end{cases} \\ c^*(G) &:= \bigcup_{n \geq 0} c^n(G). \end{aligned}$$

$c^*(G)$ を G の閉包という [8, 17].

この G^n や閉包はグラフのハミルトン性との関係はあるものの [8, 17], さほどの有用性はない. これよりも有用なのは次のような n 乗の定義ほかである [2].

E を有限集合 V 上の 2 項関係 ($E \subseteq V \times V$) とするとき, $G = (V, E)$ を有向グラフ (directed graph) といい (無向グラフは E が対称的な 2 項関係の場合である), $(u, v) \in E$ を G の (有向) 辺という. E' も V 上の 2 項関係であるとき, 和 $E \cup E'$, 共通部分 $E \cap E'$ と E の n 乗を

$$\begin{aligned} E \cup E' &:= \{(u, v) \mid (u, v) \in E \vee (u, v) \in E'\} \\ E \cap E' &:= \{(u, v) \mid (u, v) \in E \wedge (u, v) \in E'\} \\ E^n &:= \begin{cases} \{(v, v) \mid v \in V\} & (n = 0) \\ \{(u, w) \mid \exists v \text{ s.t. } (u, v) \in E \wedge (v, w) \in E\} & (n \geq 1) \end{cases} \end{aligned}$$

と定義する. $G' := (V, E')$ とするとき, $G \cup G' := (V, E \cup E')$ を G と G' の和, $G \cap G' := (V, E \cap E')$ を G と G' の共通部分といい,

$$G^n := (V, E^n) \tag{2.11}$$

を G の n 乗という. また,

$$G^{\leq n} := (V, \bigcup_{i \leq n} E^i), \quad G^+ := (V, \bigcup_{i \geq 1} E^i), \quad G^* := (V, \bigcup_{i \geq 0} E^i)$$

と定義し, G^+ , G^* をそれぞれ G の推移閉包 (transitive closure), 反射推移閉包 (reflexive transitive closure) という.

E' と G' および $G \cup G'$, $G \cap G'$ は容易に $E' \subseteq V' \times V'$, $G' := (V', E')$,

$$G \cup G' := (V \cup V', E \cup E'), \tag{2.12}$$

$$G \cap G' := (V \cap V', E \cap E') \tag{2.13}$$

と一般化でき, $G \cup G'$, $G \cap G'$ は上記 (4) で定義した 2 つのグラフの和と共通部分に一致する:

命題 2.2. (1) 2 つのグラフの和の定義 (2.5) と (2.12) は一致する.

(2) 2 つのグラフの共通部分の定義 (2.6) と (2.13) は一致する.

(3) G^n の 3 つの定義 (2.9), (2.10), (2.11) は異なる.

G^n , $G^{\leq n}$, G^+ , G^* の隣接行列は G の頂点間の距離や到達可能性 (reachability; 道の有無) を表すことができるなど, 有用である.

命題 2.3. グラフ G の位数を p , 頂点集合を $V(G) = \{v_1, \dots, v_p\}$ とし, 隣接行列を $A(G) = (a_{ij})$ で表す. 次が成り立つ.

- (1) $A(G^n)$ において, $a_{ij} = 1 \iff v_i$ から v_j への長さ n の道が存在する.
- (2) $A(G^{\leq p-1})$ において, $a_{ij} = 1 \iff v_i$ から v_j へ到達可能である.
- (3) $A(G^*)$ において, $a_{ij} = 1 \iff v_i$ から v_j へ到達可能である.

証明 (1) 2 項関係 E の n 乗 E^n の定義より,

$$(u, v) \in E^n \iff u \text{ から } v \text{ へ長さ } n \text{ の道が存在する}$$

が成り立つことが n に関する帰納法で容易に証明できる.

(2) は 2 頂点間の距離はたかだか $p-1$ であることより, (3) は (2) より導かれる. \square

G_1 と G_2 の直積 $G_1 \times G_2$ では G_1 の頂点としての隣接性と G_2 の頂点としての隣接性をそれぞれ独立に考慮して $G_1 \times G_2$ の頂点間の隣接性を定義したが, これに加えて G_1 での隣接と G_2 での隣接の推移性も考慮して辺を加えたものを強積 (strong product) といい [26], $G_1 \boxtimes G_2$ とか $G_1 * G_2$ 等で表す:

$$G_1 \boxtimes G_2 := (V_1 \times V_2, \{(u_1, u_2)(v_1, v_2) \mid (u_1 v_1 \in E_1 \wedge u_2 = v_2) \vee (u_2 v_2 \in E_2 \wedge u_1 = v_1) \\ \vee (u_1 v_1 \in E_1 \wedge u_2 v_2 \in E_2)\}).$$

強積は古くから知られている演算であるがグラフ理論の教科書ではほとんど取り上げられていなかった. しかし最近, 小直径グラフの研究において注目され, 強積をベースにしてさらに頂点を置換する多重星積 (multiple-star product) という演算を用いて直径の小さいグラフを構成することに成功している [1, 4].

強連結に関しては, n 乗 $G^{\boxtimes n}$ も次のように自然に定義できる:

$$G^{\boxtimes n} := \overbrace{G \boxtimes \cdots \boxtimes G}^n.$$

(7) 商と余り

四則演算のうち和, 差, 積の定義があるのに対し, 商の定義はない. 必要性が薄いので当然ではあるが, あえて導入するのであれば, 積 (直積) の逆演算になるように定義するのがよい. G_1 を G_2 で割った商 G_1/G_2 と余り $G_1 \bmod G_2$ を次のように定義する.

$$\begin{aligned} G_1/G_2 &:= G_3 \xLeftrightarrow{\text{def}} \exists G'_1 \subseteq G_1 \text{ s.t. } G_3 \cong G'_1 \times G_2 \\ &\quad (\text{ただし, } G'_1 \text{ は } G_1 \text{ の極大部分グラフ}), \\ G_1 \bmod G_2 &:= G_4 \xLeftrightarrow{\text{def}} G_4 \cong (G_1 - G_1/G_2). \end{aligned}$$

$G'_1 \cong G$ のとき, G_1 は G_2 で割り切れると定義する.

残念ながら, この定義だと完全な逆演算にはならない. 因みに, 頂点や辺に名称が付いていないグラフ G_1, G_2 の場合にも一般には $(G_1 - G_2) + G_1 \cong G_2$ は成り立たないので, $-$ は $+$ の逆演算ではない.

命題 2.4. (1) G_1/G_2 は、同型を除き一意的に定まる.

(2) $(nG_1)/G_2 \cong n(G_1/G_2)$.

(8) 並列化と環状化

並列化と環状化は直積の特別の場合である. すなわち, $G \times P_n$ を G の並列化 (parallelization) といい, $G \times C_n$ を G の環状化 (cyclization) と呼ぶことにする. 並列化や環状化によって表されるグラフは, 元になるグラフの性質が保たれるか否かを考えやすいため, グラフ理論の初心者クラスの演習問題を作るのに役立つ. 例えば, G がハミルトングラフ/オイラーグラフ/2部グラフのとき, その並列化グラフや環状化グラフが再びハミルトングラフ/オイラーグラフ/2部グラフになるための条件を考えることは適度なむずかしさをもつ好例である:

命題 2.5. (1) G がハミルトングラフならば $G \times P_n$ もハミルトングラフである.

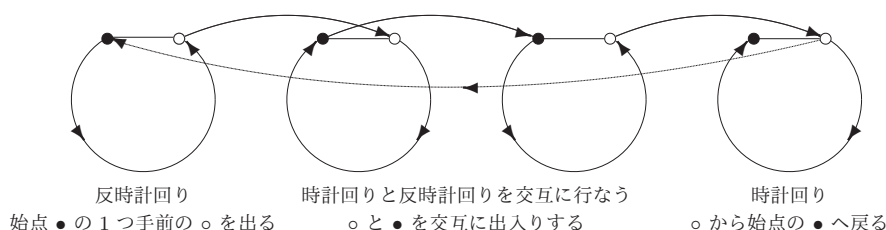
(2) $n \geq 3$ のとき, $G \times C_n$ がオイラーグラフである必要十分条件は G がオイラーグラフであることである.

(3) G が2部グラフである必要十分条件は $G \times P_n$ が2部グラフであることである.

証明(概略)

(1) $n = 1$ の場合, 自明.

$n \geq 2$ の場合, n が偶数の場合, 下図のようなハミルトン閉路がある. 円状のサイクルは G のハミルトン閉路.



n が奇数のときはもっと複雑で, 最初の G では, ハミルトン閉路の終点の2つ手前を出る. 以後, 時計回りと反時計回りを交互に繰り返す (途中の G のハミルトン閉路では, 始点の1つ手前と2つ手前を交互に出入りし), 最後の G においては, 始点の1つ手前を出て最初の G の始点の1つ手前に戻ったら, そこから最初の始点に戻ればよい.

(2) $G \times C_n$ においては, どの頂点も次数が2ずつ増えるから.

(3) $G \times P_n$ において生じる新しいサイクルは, G の異なるコピーの同じ部の側の頂点同士を結ぶ辺と G の辺を経て, G の異なるコピーの反対側の部の頂点同士を結ぶ辺を経て, 最後に G の辺を経て始点に戻るので, たどる辺の本数は必ず偶数である. \square

並列化グラフあるいは環状化グラフとして表すことのできる事物の具体例は5節で述べる.

(9) 代入

与えられたグラフの特定の頂点を指定して, そこに別のグラフを代入することを考えよう. この‘代入’という演算の特別な場合として, $G_1 = (V_1, E_1)$ のすべての頂点に同じグラフ $G_2 = (V_2, E_2)$

を代入する演算 $G_1[G_2]$ は合成 (composition) とか辞書式積 (lexicographic product) という名称で知られている [3, 14, 17, 24] :

$$G_1[G_2] := (V_1 \times V_2, \{(u_1, u_2)(v_1, v_2) \mid u_1v_1 \in E_1 \vee ((u_1 = v_1) \wedge (u_2v_2 \in E_2))\}).$$

合成 $G_1[G_2]$ では G_1 のすべての頂点に G_2 のコピーを代入するが, 特定の頂点だけを指定して代入するように変えた方がより一般的な演算になる. まず, 基本となる 1 点への代入 (substitution) を次のように定義する [3]. $G_1 = (V_1, E_1)$, $v_1 \in V_1$, $G_2 = (V_2, E_2)$ に対して, $G_1[v_1, G_2]$ を

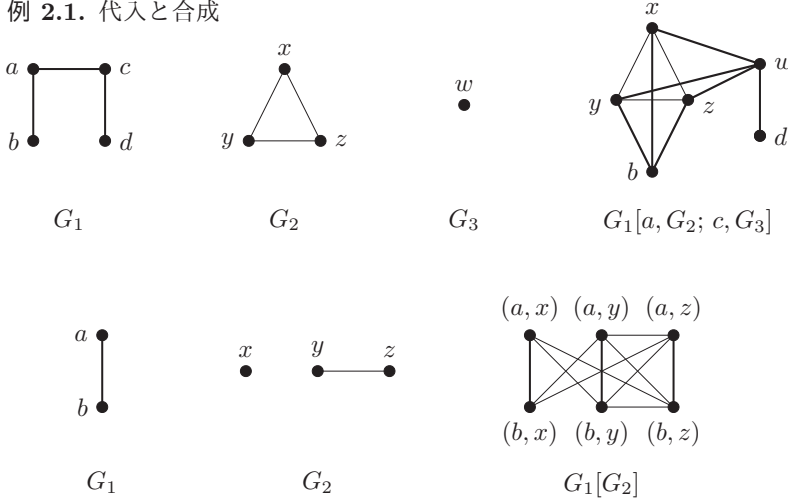
$$G_1[v_1, G_2] := (V_1 \cup V'_2, E_1 \cup E'_2 \cup \{u_1v'_2 \mid u_1v_1 \in E_1, v'_2 \in V'_2\})$$

と定義する. ここで, $V'_2 := \{v' \mid v \in V_2\}$, $E'_2 := \{u'v' \mid uv \in E_2\}$ であり, $G'_2 := (V'_2, E'_2)$ は $V_1 \cap V'_2 = \emptyset$ となるようにした G_2 のコピーである. この演算は, まず, G_1 を描き, 頂点 v_1 の所に G'_2 を代入するように描き (G_1 の辺は, v_1 に接続しているもの以外はそのまま残す), G'_2 の各頂点と, v_1 が接続していた G_1 の各頂点とを辺で結ぶという演算である.

さらに一般的に, v_1, \dots, v_n をグラフ G の頂点とすると, v_i にグラフ G_i を代入 ($i = 1, \dots, n$) して得られるグラフ $G[v_1, G_1; \dots; v_n, G_n]$ を, 次のように定義する [3] :

1. まず, G を描き,
2. その頂点 v_i の所に G_i を代入するように描く ($i = 1, \dots, n$).
3. G の辺は, v_1, \dots, v_n に接続しているもの以外はそのまま残す.
4. v_i ($i = 1, \dots, n$) が接続していた G の各頂点と G_i の各頂点を辺で結ぶ.
5. $v_iv_j \in E$ ならば G_i の各頂点と G_j の各頂点を辺で結ぶ.

例 2.1. 代入と合成



命題 2.6. $V(G) = \{v_1, v_2, \dots, v_p\}$, $\{u_1, u_2, \dots, u_k\} \subseteq V(G)$ とする.

- (1) $G[u_1, G_1; u_2, G_2; \dots; u_k, G_k] \cong (\dots((G[u_1, G_1])[u_2, G_2])\dots)[u_k, G_k]$.
- (2) $G[G'] \cong G[v_1, G'; v_2, G'; \dots; v_p, G']$. すなわち, 合成 $G[G']$ は, G のすべての頂点に G' を代入して得られるグラフである.

証明 (1) は自明とは言い難いので, k に関する帰納法で証明する. $k = 1$ のときは明らか. $k > 1$ のとき, $H_i := (\cdots ((G[u_1, G_1])[u_2, G_2]) \cdots) [u_i, G_i]$ とおく. まず, $u_k \in V(H_{k-1})$ すなわち u_k は G の頂点として H_{k-1} の中にまだ残っていることに注意する. さらに, $u_k v \in E(G)$ のとき, (i) $v \notin \{u_1, \dots, u_{k-1}\}$ ならば $v \in V(H_{k-1})$ であり, (ii) $v = u_j$ ($1 \leq j \leq k-1$) ならば帰納法の仮定から任意の $v'_j \in V(G'_j)$ に対して $u_k v'_j \in E(G'_j) \subseteq E(H_{k-1})$ である. (i) の場合には, 上記の構成法の 4. によって H_{k-1} に追加されるべき頂点と辺が生じるがそれらは, H_{k-1} においてその部分グラフたち G'_1, \dots, G'_{k-1} とは独立に $H_{k-1}[u_k, G_k]$ の辺として増減なく定まる. 一方, (ii) の場合には, $u_k v'_j \in E(H_{k-1})$ であるから, 構成法の 5. によって H_{k-1} に追加されるべき頂点と辺が生じるが, それらはやはり $H_{k-1}[u_k, G_k]$ によって正しく定まる. また, いずれの場合も構成法の 4. あるいは 5. で生じる辺に相当するもの以外は $H_{k-1}[u_k, G_k]$ によって生成されることはない.

(2) は (1) より導かれる. □

系 2.1. 代入 $G[u_1, G_1; u_2, G_2; \dots; u_k, G_k]$ は u_1, u_2, \dots, u_k の適用順によらない.

(10) 細分と縮約

グラフ G から辺 uv を除去し, 代りに 1 点 w と 2 辺 uw, vw を付け加えて得られるグラフを G' とするとき, この操作を $G \rightarrow G'$ と書くことにする. G' を G の初等細分 (elementary subdivision) という. これは, 辺 uv の中間に新しい頂点 w を挿入する操作 (グラフ上の 2 項演算) である. \rightarrow の反射推移閉包 \rightarrow^* は初等細分を 0 回以上繰り返す操作であり^{†1}, $G \rightarrow^* G'$ のとき, G' を G の細分 (subdivision) という [8, 11, 17, 24].

細分はグラフの平面性を特徴付ける Kuratowski の定理 [27] で使われ, [17, 24] では

$$G_1 \cong G_2 \text{ または } \exists G \text{ s.t. } (G \rightarrow^* G_1 \wedge G \rightarrow^* G_2) \quad (2.14)$$

が成り立つとき G と G' は位相同型 (homeomorphic) であると定義している (名称こそないが, [11, 19] でも同じ概念を用いている).

\rightarrow の逆演算 \rightarrow^{-1} は, G' に 2 辺 uw, vw があり $\deg(w) = 2$ であるとき, G' からこれらの 2 辺を除去し, 代りに辺 uv を付け加えたグラフを得る演算であるが, 本論文ではこれを初等辺短縮 (elementary edge-reduction) と呼び, その反射推移閉包を辺短縮 (edge-reduction) と呼ぶことにする.

$$G \rightleftharpoons G' \stackrel{\text{def}}{\iff} G \rightarrow G' \text{ または } G' \rightarrow^{-1} G$$

と定義すると, \rightleftharpoons はグラフ上の対称的な 2 項関係であり, その反射推移閉包 \rightleftharpoons^* は同値関係である. [3] では,

$$G_1 \cong G_2 \text{ または } \exists G \text{ s.t. } (G \rightleftharpoons^* G_1 \wedge G \rightleftharpoons^* G_2) \quad (2.15)$$

であるとき, G_1 と G_2 は位相同型であると定義している. (2.14) よりも (2.15) の方が自然な定義であると思われるが, 実際, 両者は同値である:

^{†1} 一般に, R を A 上の 2 項演算 ($R \subseteq A \times A$) とするとき, $\bigcup_{n \geq 0} R^n$ を R の反射推移閉包という.

命題 2.7. (2.14) と (2.15) は同値である.

証明 (2.14) \implies (2.15) は明らか.

(2.15) \implies (2.14) : $\exists G$ s.t. $(G \rightrightarrows^* G_1 \wedge G \rightrightarrows^* G_2)$ とする. $G \rightrightarrows^* G_1$ のとき, G_1 は G のいくつかの辺を細分したり辺短縮したりして得られるグラフである (厳密な証明は \rightrightarrows と \rightrightarrows^{-1} の適用回数に関する帰納法で示すことができるが, 明らかであろう). $G \rightrightarrows^* G_2$ についても同様である.

$e = uv$ を G の辺とする. $G \rightrightarrows^* G_1$ を満たす G_1 を得るために適用した $\rightrightarrows, \rightrightarrows^{-1}$ のうち, 辺 e に適用されたものを適用順に

$$(\rightrightarrows)^{m_1} (\rightrightarrows^{-1})^{n_1} \dots (\rightrightarrows)^{m_k} (\rightrightarrows^{-1})^{n_k} \quad (k \geq 0)$$

とする. ここで, 上付き添え字 m_i, n_i (≥ 0) は連続した適用回数を表す. $n_1 \leq m_1$ および, 任意の i について, $n_i \leq \sum_{j=1}^{i-1} (m_j - n_j) + m_i$ が成り立っていないといけない. 同様に, $G \rightrightarrows^* G_2$ についても

$$(\rightrightarrows)^{m'_1} (\rightrightarrows^{-1})^{n'_1} \dots (\rightrightarrows)^{m'_{k'}} (\rightrightarrows^{-1})^{n'_{k'}} \quad (k' \geq 0)$$

とする. $(G \rightrightarrows^* G_1 \wedge G \rightrightarrows^* G_2)$ であることより, $e \in E(G)$ に

$$l_e := \sum_{j=1}^k (m_j - n_j) = \sum_{j=1}^{k'} (m'_j - n'_j)$$

回の初等細分が適用すると G_1 の辺 e_1 と G_2 の辺 e_2 が得られ, $e_1 \cong e_2$ である. このことが任意の $e \in E(G)$ に対して成り立っているので, $G \rightrightarrows^* G_1 \wedge G \rightrightarrows^* G_2$ が成り立っている. \square

\rightrightarrows^{-1} は初等縮約と呼ばれる操作の特別な場合である. 初等縮約 (elementary contraction) とは, 任意の隣接する頂点 u と v を同一視する操作 (u と v の中間に $\deg(w) = 2$ なる頂点 w がちょうど 1 つだけあるという制約は設けない) である. 厳密な定義は省略するが, 単純グラフだけを考えるか, あるいは多重ループや多重辺も許した多辺グラフを考えるかによって, 初等縮約によって得られたグラフから自己ループを削除したり多重辺を 1 辺だけに置き換えたりするか否かの違いがある. 初等縮約を 0 回以上行なってグラフ G からグラフ G' が得られるとき, G' を G の縮約 (contraction) という [17, 24]. H' が G の部分グラフ H の縮約であるとき, H' を G の縮約部分グラフ (subcontraction) という.

(11) 双対グラフ

$G = (V, E)$ を平面グラフとすると, その双対 (dual) は次のように定義されるグラフ G^* である:

$$G^* := (\{r^* \mid r \text{ は } G \text{ の領域}\}, \{r_1^* r_2^* \mid e \text{ は } r_1 \text{ と } r_2 \text{ の境界をなす辺}\}).$$

一般に, G^* は多重辺や自己ループをもちうる多辺グラフになる.

双対グラフが領域に関して定義されているのに対して, そのアナロジーとして, G の辺双対グラフ (edge-dual graph) \tilde{G} を次のように定義することができる:

$$\tilde{G} := (\{\tilde{e} \mid e \in E\}, \{\tilde{e}_1 \tilde{e}_2 \mid e_1, e_2 \in E, e_1 \text{ と } e_2 \text{ は隣接する}\}). \quad (2.16)$$

ここで, e_1 と e_2 が隣接するとは端点を共有することである. この定義だと, G が単純グラフであっても \tilde{G} のどの頂点も自己ループをもつが多重辺は生じないので, 定義 (2.16) を $e_1 \neq e_2$ と制限する. よって, G が単純グラフなら \tilde{G} も単純グラフである.

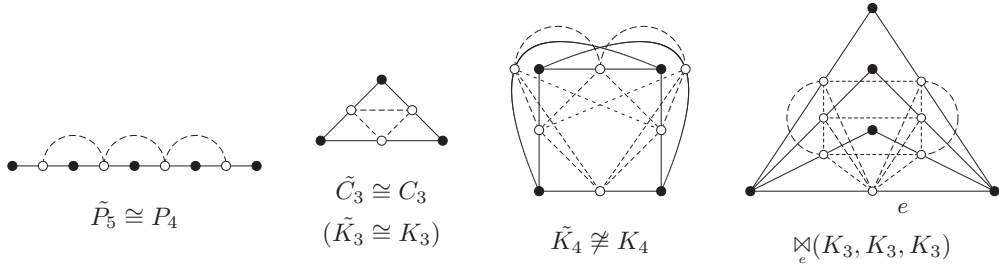
例 2.2. (1) $n \geq 2$ のとき, $\tilde{P}_n \cong P_{n-1}$ である (P_n は位数が n の道). よって, 辺双対の辺双対は必ずしも元のグラフに戻らない. それゆえ, 辺‘双対’という用語は適切とはいえないかもしれない.

(2) G がサイクル C_n の場合, $\tilde{G} \cong G$ である. このような場合, G は辺自己双対 (self-edge-dual) であるということにする.

(3) $n \neq 3$ ならば, 完全グラフは辺自己双対ではない: $\tilde{K}_n \not\cong K_n$.

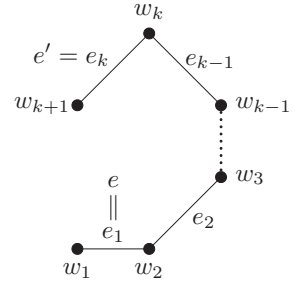
(4) G が平面グラフであっても \tilde{G} は平面的グラフであるとは限らない. K_3 の任意の 1 辺を e とするとき, 3 つの K_3 をそれぞれの e で貼り合わせたグラフ $\boxtimes_e(K_3, K_3, K_3)$ はその一例である ($\boxtimes_e(K_3, K_3, K_3)$ の辺双対は K_5 を縮約部分グラフとして含むので平面的グラフではない [8](Theorem 6.18). 演算 \boxtimes の定義については 3 節で述べる.

以下の図において, \circ は辺双対グラフの頂点を表し, 破線は辺を表す.



命題 2.8. G が連結ならば \tilde{G} も連結である. 逆は成り立たない.
証明 G を連結とし, $e = uv, e' = u'v'$ を G の任意の 2 辺 (\tilde{e}, \tilde{e}' は \tilde{G} の頂点) とする. G は連結だから, u および v から u' および v' への道が存在する. それらのうちで距離が最大のものを w_1, w_2, \dots, w_{k+1} ($k \geq 2$) とする. 辺 $w_i w_{i+1}$ を e_i ($1 \leq i \leq k$) で表す. $e_1 = e, e_k = e'$ であり, 各 i ($1 \leq i \leq k-1$) について, e_i と e_{i+1} は隣接している. よって, $\tilde{e}_i \tilde{e}_{i+1} \in E(\tilde{G})$ ($1 \leq i \leq k-1$) である. すなわち, \tilde{e} と \tilde{e}' は \tilde{G} において連結である.

逆が成り立たない一例は $2P_2$.



G が辺自己双対であるためには G の位数とサイズが等しいことが必要条件であるから, G はかなり疎なグラフである.

(12) 論理演算

論理積 (conjunction)

$$G_1 \wedge G_2 := (V_1 \times V_2, \{(u_1, u_2)(v_1, v_2) \mid u_1 v_1 \in E_1 \text{ かつ } u_2 v_2 \in E_2\})$$

は [25] で導入され, 次のことが知られている:

命題 2.9. [24, 25]

(a) $G_1 \times G_2 \cong G_1 \wedge G_2 \iff G_1 \cong G_2 \cong C_{2n+1}$.

(b) G_1, G_2 が連結のとき, $G_1 \wedge G_2$ が連結 $\iff G_1$ または G_2 が奇サイクルをもつ.

論理積のアナロジーとして論理和 (disjunction) を

$$G_1 \vee G_2 := (V_1 \times V_2, \{(u_1, u_2)(v_1, v_2) \mid u_1 v_1 \in E_1 \text{ または } u_2 v_2 \in E_2\}) \quad (2.17)$$

と定義しよう. 命題 2.9 に対応する結果として, 次のことが成り立つ:

命題 2.10. (1) $G_1 \times G_2 \subsetneq G_1 \vee G_2$, すなわち $G_1 \times G_2$ は $G_1 \vee G_2$ の部分グラフであり, かつ $G_1 \times G_2 \not\cong G_1 \vee G_2$ となる G_1, G_2 が存在する.

(2) $F_1, F_2 \subseteq V_1 \times V_2$ を

$$F_1 := \{(u_1, x_2)(v_1, y_2) \mid u_1 v_1 \in E_1, x_2, y_2 \in V_2 (x_2 \neq y_2)\},$$

$$F_2 := \{(x_1, u_2)(y_1, v_2) \mid u_2 v_2 \in E_2, x_1, y_1 \in V_1 (x_1 \neq y_1)\}$$

と定義すると, $G_1 \vee G_2 \cong (G_1 \times G_2) + (F_1 \cup F_2)$ である.

(3) G_1, G_2 が連結ならば $G_1 \vee G_2$ も連結である.

証明 (1) $G_1 \times G_2 \subsetneq G_1 \vee G_2$ は定義より明らか. また, $G_1 \times G_2 \not\cong G_1 \vee G_2$ となる G_1, G_2 も容易に示すことができる.

(2) $G_1 \times G_2$ の定義 (2.8) と $G_1 \vee G_2$ の定義 (2.17) を見比べると, $F_1 \cup F_2$ は $G_1 \vee G_2$ にあるが $G_1 \times G_2$ にはない辺の集合に等しいことがわかる.

(3) $(u_1, u_2), (v_1, v_2) \in V_1 \times V_2$ とする. まず, G_1 も G_2 も連結だとすると $G_1 \times G_2$ も連結であることは容易に示すことができる. 次に, $G_1 \times G_2$ が連結だとすると, $G_1 \times G_2$ において (u_1, u_2) から (v_1, v_2) への道

$$(u_1, u_2) = (w_1^{(1)}, w_2^{(1)}), (w_1^{(2)}, w_2^{(2)}), \dots, (w_1^{(n)}, w_2^{(n)}) = (v_1, v_2) \quad (n \geq 1) \quad (2.18)$$

が存在し, 各 i ($1 \leq i \leq n-1$) に対し $w_1^{(i)} w_1^{(i+1)} \in E_1 \wedge w_2^{(i)} = w_2^{(i+1)}$ または $w_2^{(i)} w_2^{(i+1)} \in E_2 \wedge w_1^{(i)} = w_1^{(i+1)}$ が成り立っている. よって, $w_1^{(i)} w_1^{(i+1)} \notin E_1$ ならば $w_2^{(i)} w_2^{(i+1)} \in E_2$ であり, $w_2^{(i)} w_2^{(i+1)} \notin E_2$ ならば $w_1^{(i)} w_1^{(i+1)} \in E_1$ であるから, $w_1^{(i)} w_1^{(i+1)} \in E_1 \vee w_2^{(i)} w_2^{(i+1)} \in E_2$ が成り立っている. これは $(w_1^{(i)}, w_2^{(i)})(w_1^{(i+1)}, w_2^{(i+1)}) \in E(G_1 \vee G_2)$ を意味するので, (2.18) は $G_1 \vee G_2$ における道である. ゆえに, $G_1 \vee G_2$ は連結である. \square

系 2.2. $G_1 \vee G_2 \cong G_1 \times G_2 \iff F_1 \cup F_2 = \emptyset$.

オープン問題 2.1. 論理演算を定義するのであれば, $\neg(\neg G) \cong G$, $\neg(G_1 \wedge G_2) \cong \neg G_1 \vee \neg G_2$, $\neg(G_1 \vee G_2) \cong \neg G_1 \wedge \neg G_2$ 等が成り立つような自然な演算 \neg, \wedge, \vee が定義できるとよいが, 可能であろうか?

3 貼り合せと置換

operation には「操作」「演算」という意味以外に「手術」という意味もあるくらいであり, グラフ上の演算も切り貼りである. しかし, これまでに述べた切り貼り方法では特定の箇所に特定のものを代入することはできたが, 特定の箇所同士を貼り合わせる (ペーストする) 操作や, 頂点や辺や部分グラフの入れ替え (置換) といった操作はなかった. ここではまず, 特定の箇所と

して頂点や辺を考え、頂点同士あるいは辺同士を貼り合わせる演算を考える。例えば、例 2.2 の (4) で述べた \bowtie は辺同士を貼り合わせる演算であり、次のように定義できる。

(13) 辺のペースト

e_i を G_i の辺とする ($i = 1, 2$)。 G_1 と G_2 を辺 e_1 と e_2 を同一視することによって貼り合わせたグラフを $G_1 \underset{e_1=e_2}{\bowtie} G_2$ で表す。貼り合わせて同一視した辺を e_3 としたい場合には $G_1 \underset{e_3 \leftarrow (e_1=e_2)}{\bowtie} G_2$ と表す。一般に、 $e_1 = u_1 v_1$, $e_2 = u_2 v_2$ とし、 $G' = (V', E')$ を $G = (V, E)$ のコピー、すなわち $V' := \{v' \mid v \in V\}$, $E' := \{u'v' \mid uv \in E\}$, $G' := (V', E')$ とする ($v \in V$ に対し、 v' は新しい頂点) とし、 $G_1 \underset{e_3 \leftarrow (e_1=e_2)}{\bowtie} G_2$ を次のように定義する：

$$\begin{aligned} G_1 \underset{e_3 \leftarrow (e_1=e_2)}{\bowtie} G_2 &:= (V_3, E_3), \\ V_3 &:= (V_1 - \{u_1, v_1\}) \cup (V'_2 - \{u'_2, v'_2\}) \cup \{u, v\}, \\ E_3 &:= (E_1 - e_1) \cup (E_2 - e_2) \cup \{e_3\} \\ &\quad \cup \{uw_1, vw_1, uw'_2, vw'_2 \mid u_1 w_1 \in E_1, v_1 w_1 \in E_1, u'_2 w'_2 \in E'_2, v'_2 w'_2 \in E'_2\}. \end{aligned}$$

e_3 を省略する場合には、単に $G_1 \underset{e_1=e_2}{\bowtie} G_2$ と表す。さらに、複数の辺同士を同時に貼り合わせる場合 (例えば、 $e_{1,1} \in E_1$ と $e_{2,1} \in E_2$ および $e_{1,2} \in E_1$ と $e_{2,2} \in E_2$ を貼り合わせる場合) には、

$$G_1 \underset{e_{1,1}=e_{2,1} \wedge e_{1,2}=e_{2,2}}{\bowtie} G_2$$

と表す。貼り合わせる辺の対が多数の場合も同様に定義する。

(14) 頂点のペースト

頂点と頂点を貼り合わせることも同様に定義できる：

$$\begin{aligned} G_1 \underset{v \leftarrow (v_1=v_2)}{\bowtie} G_2 &:= (V_3, E_3), \\ V_3 &:= (V_1 - \{v_1\}) \cup (V'_2 - \{v'_2\}) \cup \{v\}, \quad E_3 := \{vw_1, vw'_2 \mid v_1 w_1 \in E_1, v'_2 w'_2 \in E'_2\}. \end{aligned}$$

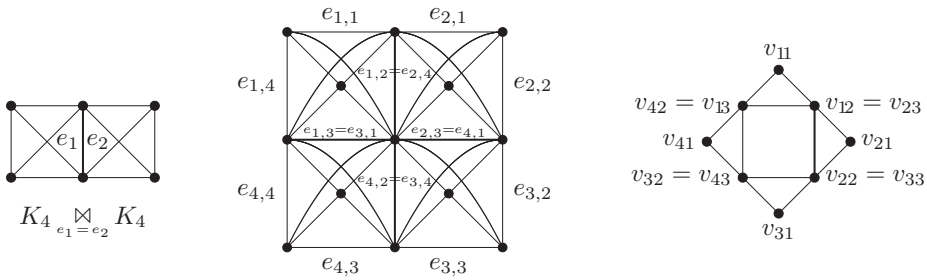
例 3.1. 貼り合わせ演算は、例 2.2 の (4) や下記の例からわかるように、新しいグラフを定義する方法として有用である。左下図に示したグラフは 2 つの K_4 をそれぞれの任意の 1 辺 e_1 と e_2 を共有するように貼り合わせたものであり、 $K_4 \underset{e_1=e_2}{\bowtie} K_4$ と表すことができる。下中央図は、 K_5 においてサイクルをなすような任意の 4 辺を $e_{i,1}, e_{i,2}, e_{i,3}, e_{i,4}$ ($1 \leq i \leq 4$) とするとき、2 辺ずつを貼り合せたものである：

$$(K_5 \underset{e_{1,2}=e_{2,4}}{\bowtie} K_5) \underset{e_{1,3}=e_{3,1} \wedge e_{2,3}=e_{4,1}}{\bowtie} (K_5 \underset{e_{4,2}=e_{3,4}}{\bowtie} K_5).$$

また、右下図は 4 つの C_3 の 2 頂点ずつを貼り合せたもの

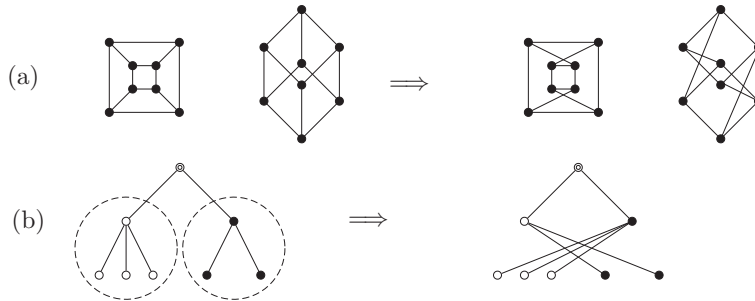
$$(C_3 \underset{v_{1,2}=v_{2,3}}{\bowtie} C_3) \underset{v_{1,3}=v_{4,2} \wedge v_{2,2}=v_{3,3}}{\bowtie} (C_3 \underset{v_{3,2}=v_{4,1}}{\bowtie} C_3)$$

であるが、 C_4 の各辺に C_3 の 1 辺を貼り合せたものとも見ることもできる。



(15) 置換

グラフのいくつかの頂点や辺を入れ替える演算も有用である．詳細な定義は省略するが，例えば，下図 (a) 左に示した立方体 Q_3 の垂直な 2 辺 2 組を入れ替えると下図 (a) 右のグラフが得られる．この場合，この例のように，(例えば，正則性を保つため等の理由により) 頂点の次数や直径などを変えないようにすることが多い．また，頂点の入れ替えの例を下図 (b) に示した．



辺の入れ替えは辺の加除で容易に表すことができる．頂点の入れ替えは，上図 (b) のように辺と頂点の接続性を保存するような入れ替えと，上図 (b) の破線の円同士をそっくり入れ替える方法などが考えられ，これらはいくつかの部分を入れ替える (置換する) 演算として一般化できる．ページ数の制約もあり，本論文ではこれ以上立ち入らない．

4 木に関する演算

データ構造としての木 (多くの場合，根付き木 (rooted tree) である) にはいろんな種類があるが，それらを式で表現するための適当な演算がないことが多い．木を定義するための最も一般的な方法は再帰的定義であるが，それでは定義できない木もある．例えば，系図のように，複数の子の間に何番目という順序が定まっている (順序木 (ordered tree)) が，ある子が死亡して欠けていることを表したい場合 (位置木 (positional tree)) や，ヒープ (heap) のように特殊な形をした 2 分木などである (用語についての詳細は，例えば [3] や [7] を参照のこと)．

その解決策として有効なのは形式言語を利用する方法である．[3] ではこの方法を利用して接ぎ木 (grafting) という概念を導入している．接ぎ木はペーストの特別な場合に相当する．

n を自然数とし， $1, 2, \dots, n$ それぞれを記号と考えたアルファベットを $[n] := \{1, 2, \dots, n\}$ で表す． $[n]^*$ の部分集合 \mathcal{D} が次の 3 条件を満たすとき， \mathcal{D} を (n 分木の) 樹形 (tree shape) と呼ぶ．

- (i) $\lambda \in \mathcal{D}$ (λ は根を表す)
- (ii) どの $x \in \mathcal{D}$ のどのプレフィックスも \mathcal{D} の元である (プレフィックスは先祖を表す).
- (iii) 任意の $x \in \mathcal{D}$, 任意の自然数 i に対して, $xi \in \mathcal{D}$ ならば $1 \leq j \leq i$ であるどんな j についても $xj \in \mathcal{D}$ である (xj は x の j 番目の子を表す).

例えば, 言語

$$T := \{\lambda, 1, 2, 3, 11, 12, 21, 31, 32, 33, 211, 212, 321\} \subseteq [3]^*$$

が表す樹形を右図に示した.

一般に, ℓ を樹形 $T_1 \subseteq [n]^*$ の葉とする. $x \in T_1$ かつ任意の $a \in [n]$ に対して $xa \notin T_1$ であるとき, xa を接ぎ木可能部位といい, T_1 の接ぎ木可能部位 xa において T_2 を接ぎ木して得られる木を次のように定義する:

$$T_1 \oplus_{xa} T_2 := T_1 \cup \{xat \mid t \in T_2\}.$$

さらに, [3] では, 有向グラフによって木を定義することにより, 木の様々な性質を下記の容易な命題の系として導いている.

次の条件を満たしている有向グラフ G を根付き木という:

- (i) 入次数が 0 の頂点が 1 つだけある (G の根).
- (ii) 根からどの頂点へも道がちょうど 1 つある.
- (iii) 根以外のどの頂点も入次数が 1 である.
- (iv) 出次数が 0 の頂点がいくつかある (G の葉).

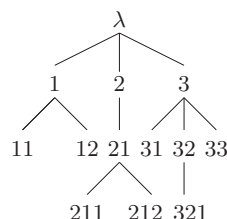
命題 4.1. [木に対する握手補題] [3] $G = (V, E)$ が木を表している有向グラフであるとき, 次の等式が成り立つ:

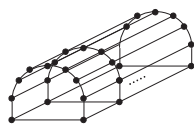
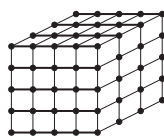
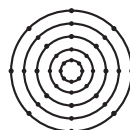
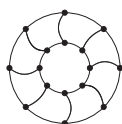
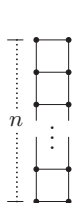
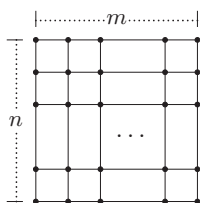
$$\sum_{v \in V} \text{in-deg}(v) = \sum_{v \in V} \text{out-deg}(v) = |V| - 1.$$

5 授業での実践効果

グラフ理論の入門クラスにおいては, いろんな演算を使って自分でいろんなグラフを定義してもらうことによってグラフへの理解をより深めることができる. ペーストや置換演算などのように, 演算自体は有用であるが, 数式が複雑になってしまい, 言葉で述べる方が簡単であるような演算もあるが, 逆に, 数式で表すと簡潔になることが多いだけでなく, 定義の厳密性 (曖昧さの排除) も担保されるというメリットがある. 実際, 小直径グラフを構成するためにグラフにいろんな置換を施してシミュレーションを行なったりする際には変換を式で (プログラムで) 表すことが必要になることが示すように, 応用上も重要である.

また, 学生にとっては, 数式で表現する能力を養うこと自体が数学力の向上に役立つ. このことは実際の授業 [2] において長年の実践を通して確認されている. ここでは, 授業 [2] や著書 [3] において提示した例や, 授業の中で学生たちに考えてもらった例をいくつか示して (ドラえもんとかラーメンといった例もあったが省略した), その報告に代える.

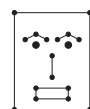


 $P_n \times C_9$ (トンネル) $P_5 \times P_4 \times P_5$ (ジャングルジム) $5C_8$ (波紋)星月夜
 $C_8 \boxplus 7C_5$ 浮輪
 $P_2 \times C_8$ 太陽
 $C_8 \boxplus 8P_2$ 梯子
 $P_2 \times P_n$ メッシュ
 $P_m \times P_n$ プリズム
 $P_2 \times K_3$ 五角柱
 $C_5 + K_1$ 車輪グラフ
 $C_n + K_1$ 星グラフ
 $K_n + K_1$ 十字架
 $2P_2$ 星
 C_5 

扇風機

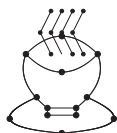
$$(((P_5 \boxtimes P_5) \boxtimes (P_3 \boxtimes P_3)) \boxtimes C_4) \boxtimes C_3$$

1頂点 1頂点 1頂点 4頂点 1頂点



顔

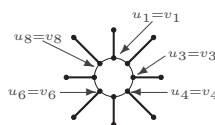
$$2C_4 \boxplus 2P_3 \boxplus 2K_1 \boxplus P_2$$



ティータイム

$$4P_3 \boxplus (((C_4 \boxtimes C_7) \boxtimes C_4) \boxtimes P_5)$$

2辺 1辺 2頂点



太陽

$$C_8 \boxtimes_{u_1=v_1 \wedge \dots \wedge u_8=v_8} 8P_2$$

[参考文献]

- [1] 水野良祐, 既知のグラフを用いた小直径グラフの構成, FIT2016 第 15 回情報科学技術フォーラムイベント企画『小直径グラフの追究〜グラフ理論の未解決問題とインターコネクトの未来〜』, 富山大学, 2016 年 9 月.
- [2] 守屋悦朗, <http://www.edu.waseda.ac.jp/~moriya/education/classes/applmath2/> (「応用数学 2」講義用サイト), 2016.
- [3] 守屋悦朗, 『ヴィジュアルでやさしいグラフへの入門』, サイエンス社, 2016.
- [4] Graph Golf — The Order/degree Problem Competition, <http://research.nii.ac.jp/graphgolf/>, since 2015.
- [5] 鈴木晋一 (編著), 『数学教材としてのグラフ理論』, 学文社, 2012.
- [6] R. Diestel, *Graph Theory*, 4th ed., Springer, 2010.
- [7] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms*, 3rd ed., The MIT Press, 2009.
- [8] G. Chartrand & L. Lesniak, *Graphs and Digraphs*, 4th ed., Chapman & Hall/CRC, 2005.
- [9] W. Kocay and D. L. Kreher, *Graphs, Algorithms, and Optimization*, Chapman & Hall/CRC, 2005.
- [10] J. L. Gross and J. Yellen (ed.), *Handbook of Graph Theory*, CRC Press, 2004.
- [11] J. Xu, *Theory and Application of Graphs*, Kluwer Academic Publishers, 2003.
- [12] G. Valiente, *Algorithms on Trees and Graphs*, Springer, 2002.
- [13] An. Biggs, *Algebraic Graph Theory*, 2nd ed., Cambridge Univ. Press, 1993.
- [14] 根上生也, 『離散構造』, 共立出版, 1993.
- [15] N. Hartsfield & G. Ringel, *Pearls in Graph Theory*, Academic Press, 1990.
- [16] A. Gibbons, *Algorithmic Graph Theory*, Cambridge Univ. Press, 1985.
- [17] M. Behzad, G. Chartrand & L. Lesniak-Foster, *Graphs and Digraphs*, Prindle, Weber & Schmidt, 1979.
- [18] B. Bollobás, *Extremal Graph Theory*, Academic Press, 1978.
- [19] C. L. Liu, *Elements of Discrete Mathematics*, McGraw-Hill, 1977.
- [20] E. Lawler, *Combinatorial Optimization, Networks and Matroids*, Dover, 1976.
- [21] A. V. Aho, J. E. Hopcroft & J. D. Ullman, *The Design and Analysis of Computer Algorithms*, Addison-Wesley, 1974.
- [22] C. W. Marshall, *Applied Graph Theory*, Wiley, 1971.
- [23] W. K. Chen, *Applied Graph Theory*, North-Holland, 1971.
- [24] F. Harary, *Graph Theory*, Addison-Wesley, 1969.
- [25] D. J. Miller, The categorical product of graphs, *Canad. J. Math.* 29, 1511–1521, 1968.
- [26] G. Sabidussi, Graph multiplication, *Math. Z.* 72, 446–457, 1960.
- [27] K. Kuratowski, Sur le problème des courbes gauches en topologie, *Fund. Math.* 15, 271–283, 1930.