講 座

設計自動化におけるグラフ理論と組み合わせ算法 (1)*

可児 賢二** 大附 辰夫***

1. はじめに

設計自動化 (DA : Design Automation) の対象分野は広範囲に及ぶので，本稿では集積回路および論理装置に対象を限定して，特にグラフ理論と密接な関連をもつ問題に焦点を絞ることとする．集積回路と論理装置は互いに大きな影響を与えながら 1960 年代はじめから技術革新をくりかえしてきており，今後も大きな変化が予想される．従って設計自動化の内容も不安定なものと考えた方が良い．このような状況のもとで定説を体系的にまとめるという講座の目的をはたすのは難題であるが，基本的な考え方は将来も何らかの形で役に立つであろうと念願して三回の連載でまとめることとした．

第一回ではグラフ理論の基礎について，第二回では DA における種々の問題の中から実用的に用いられているかというよりはむしろグラフ理論と密接な関連をもつ問題をピックアップして，それらの定式化について，そして第三回では計算機処理を前提とした算法について述べる．

2. グラフ理論の基礎

2.1 グラフ理論の背景

直観的にいえば，**グラフ** (graph) とは，いくつかの**点** (node) とそれらの点の対を両端とする線——これを**枝** (branch) と呼ぶ——とによって表わされるような図形のことである．この際，点と枝とがお互いにどのような“つながり方”をしているかだけが問題で，点の位置とか枝の長さや，曲がり方などは問題にしない．グラフを取り扱う理論，つまりグラフ理論は，古く Euler (1707-1783) の時代から研究され始め，1960 年代から電気回路，輸送問題，日程計画などの工学的応用と関連して広く研究されるようになってきた．

正統的な数学の体系からグラフ理論をながめれば，位相幾何学の中の一次元複体と呼ばれる極めて特殊で簡単な場合に過ぎずその地位は不安定なものである．しかし，理論的に“特殊な場合”というのは，“一般の場合”より具体的で詳細な議論が可能になり，能率の良い問題処理技法を確立できるかもしれないという意味で，応用の立場からみるとかえって有用であることが多い．

本講座の対象とする DA の分野にもグラフ理論が応用できる問題が数多く存在する．グラフは対象とする問題の明確な表現を与えるもので，グラフによる表現が，特定の問題の解決策を考案したり，問題の難しさを評価したりするための有効な道具となる場合が多い．さらにグラフによる表現を媒介とすることによって，計算機利用を前提とした算法やデータとの対応が明確につけられるという利点がある．しかし，グラフを扱うことによって“解けなかった問題が解けるようになる”という意味でグラフ理論が利用されるのではないことに注意されたい．

グラフ理論はこれまで広く研究されてはきたが，“定説”といえるような理論体系が確立されているとはいえない状態である．例えば，同一の概念を表わすための用語が人によってまちまちであったり，また同一の用語が人によって異なる概念を表わしていたりする．そこで，本講座で用いる用語も数多く存在するものの一つに過ぎないことに注意されたい．グラフ理論を扱った教科書に盛りこまれている内容もまちまちであるが，本講座では DA への応用を対称としたグラフ理論を紹介するのが目的であるので，DA との関連があるものは特殊な概念であってもなるべく盛りこむことにした．逆にグラフ理論の体系の上で重要な概念であっても，DA との関連の薄いものは省くこととした．

さて DA において我々が遭遇する問題の多くは，グラフの問題として表現できるが，さらに多くの問題を

---

* Graph Theory and Combinatorial Algorithms for Design Automation by Kenji Kani (IC Division, Nippon Electric Co., Ltd.) and Tatsuo Ohtsuki (Central Research Laboratories, Nippon Electric Co., Ltd.)

** 日本電気 (株) 集積回路事業部
*** 日本電気 (株) 中央研究所

扱うためには"グラフ理論"の兄弟分ともいえる"組み合わせ数学"を導入する必要がある．組み合わせ数学とはグラフの他にも整数の集合，方策の集合など有限個の離散集合を対象としたいわゆる"組み合わせ問題"の性質を論ずるものである．本稿の表題の意図するところはこのような考え方に基づいている．以下本章では一応グラフ理論に限って話を進めるが，グラフを前提とした問題の定式化や算法が，グラフで表現できない問題に対しても自然に拡張できるような説明を盛りこむよう努力した．

グラフ理論の参考書の類は数多くあるが，全般的なものとして文献 1)～7)，電気回路，通信網への応用を主眼としたものとして 8)，9)，ネットワークフローの理論を中心としたものとして 10)～13) を挙げておく．

2.2 グラフの定義と表現法

数学的な表現を用いれば，グラフ $\mathbf{G}$ は二つの集合 V および E の複合概念として定義され，$\mathbf{G}=(V,E)$ と表わされる．V の元 v は**節点**と呼ばれる．E の元 $e=(u,v)$ は二つの節点 u，$v\in V$ の対を表わし，これを**枝**と呼ぶ．以下の議論においては，V，E が共に有限集合となる場合を対象とする．グラフと似た意味を持つ概念として"ネットワーク"という言葉があるが，これはグラフにおいて節点や枝と具体的な物との対応とか，節点や枝に割り当てられた物理量を含んだ概念であり，"グラフ"とは区別するのが普通である．例えば**図-1**(a)の電気回路網はネットワークであり，これから物理的意味を取り除いて抽象化すれば，同図(b)のようなグラフとなる．

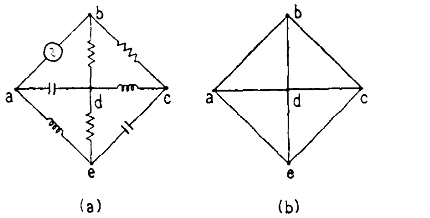

図-1 電気回路網(a)とそのグラフ表現(b)

一つの枝 $e=(u,v)$ に対して，u，vを枝 e の**端点**という．特に $u=v$ の場合には，枝 e は**セルフループ**と呼ばれる．また二つ以上の枝が同一の節点対を端点としてもつ場合，これらを**並列枝**という．一方，一つの節点 v に対して枝 e が v を端点としてもつとき，枝 e は節点 v に**接続している**という．また特定の節点に接続している枝の数をその節点の**次数**(degree)という．二つの節点 u，v の間に枝が存在するとき，節点 u，v は互いに**隣接している**という．**図-2**のグラフを例にとれば，枝 e_4 と e_5 は並列枝，枝 e_1 はセルフループである．枝 e_6 に着目すれば，その端点は $\{v_3, v_4\}$ である．節点 v_3 に着目すれば，これに接続している枝は $\{e_4, e_5, e_6\}$――従って v_3 の次数は 3――であり，隣接している節点は v_1，v_4 である．

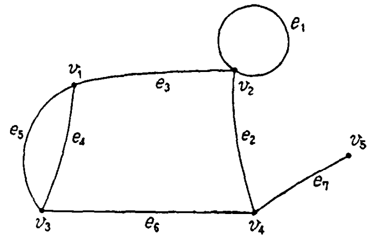

図-2 グラフの例

特別な構造を有するグラフには特別な名前が付けられ，その各々の性質が一般のグラフよりも深く論じられている場合がある．これらについては順を追って説明するが，まず代表的なものとして完全グラフと2部グラフについて述べておこう．セルフループがなく，かつ任意の異なる2節点を両端点とする枝がちょうど一個存在するようなグラフを**完全グラフ** (complete graph)という．完全グラフ $\mathbf{G}=(V,E)$ の節点の数，枝の数をそれぞれ $n=|V|$，$m=|E|$ とすれば（ここで $|\cdot|$ は集合の元の数を表わす），$2m=n(n-1)$ という関係が成立することは明らかである．また節点の集合 V が二つの部分 V_1，V_2 に分割 ($V_1\cap V_2=\phi$, $V_1\cup V_2=V$) されて，全ての枝 $e\in E$ の端点の一方が V_1 に他方が V_2 に属する，という性質を有するグラフを**2部グラフ**(bipartite graph)という．**図-3**(a)，(b)のグラフはそれぞれ完全グラフ，2部グラフの例である．

グラフの接続関係を行列の形で表現しておくと便利なことがある．まずグラフ $\mathbf{G}=(V,E)$ の節点対に着

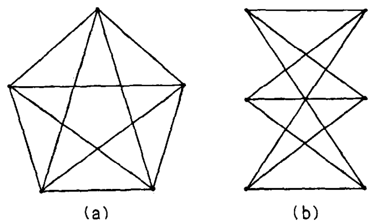

図-3 クラトフスキ・グラフ

目した表現法として，

$$A=\{a_{ij}\}\ ;\ a_{ij}=\begin{cases}1:(i,j)\in E\\0:(i,j)\notin E\end{cases} \tag{2・1}$$

で定義される対称行列を**隣接行列** (adjacency matrix) という．隣接行列を用いた場合，並列枝を表現することができない．そこで各々の節点対の間の枝の数を行列の要素として定義することもある．例えば図-2 のグラフの隣接行列は下記の通りである．

$A=$ (2・2)

V \ V	1	2	3	4	5
1	0	1	2	0	0
2	1	1	0	1	0
3	2	0	0	1	0
4	0	1	1	0	1
5	0	0	0	1	0

次に枝ごとの端点に着目した表現法として，

$$D=\{d_{ij}\}\ ;\ d_{ij}=\begin{cases}1:\text{節点 } i \text{ が枝 } j \text{ の端点のとき}\\0:\text{その他の場合}\end{cases} \tag{2・3}$$

で定義される**接続行列**(incidence matrix) を用いることもある．例えば図-2 のグラフの接続行列は

$D=$ (2・4)

V \ E	1	2	3	4	5	6	7
1	0	0	1	1	1	0	0
2	1	1	1	0	0	0	0
3	0	0	0	1	1	1	0
4	0	1	0	0	0	1	1
5	0	0	0	0	0	0	1

となる．グラフがセルフループを含まない場合，接続行列の各列はちょうど二つの“1”要素をもつという性質がある．そこで，一つの行を取り除いた行列 $\hat{D}$ (これを既約接続行列という)を与えても，取り除かれた行を再現することができる．既約接続行列は電気回路解析において重要な役割を果たす (3.2 節参照)．

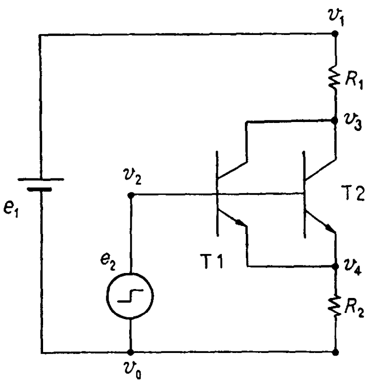

図-4　トランジスタ回路

接続行列は特に電気回路の接続関係を表わすのに適している．すなわち抵抗，キャパシタ，電源などの 2 端子素子を列に，等電位点を行に対応させることによって極めて自然な表現ができるからである．**図-4** のように 3 端子素子（トランジスタ）を含んでいても，一つの列に三つ以上の“1”要素を含むことを許すように接続行列の意味を拡張解釈すれば，図-4 は，

$D=$ (2・5)

V \ E	T_1	T_2	R_1	R_2	e_1	e_2
0	0	0	0	1	1	1
1	0	0	1	0	1	0
2	1	1	0	0	0	1
3	1	1	1	0	0	0
4	1	1	0	1	0	0

と表現できる．ところが図-1 のように 2 端子素子のみから成る回路と違って，これに対応したグラフを描くことはできない*．そこで 3 個以上の端点をもつことができるような（広義の）枝を含むグラフを**ハイパーグラフ** (hypergraph) と定義して，グラフ理論を一般の組合せ問題に拡張しようという試みもある[4][14]．

前に述べたように，グラフにおいては節点の位置とか枝の長さは問題としない．そこで実際の応用では二つのグラフが同じものであるか否かが問題となることがある．グラフ $\mathbf{G}_1$ と $\mathbf{G}_2$ が同数個の節点と同数個の枝を有し，両者の節点の集合の間，枝の集合の間に適当な一対一対応が存在して，その対応のもとで両者の接続関係が等しくなるとき，$\mathbf{G}_1$ と $\mathbf{G}_2$ は**同形**(isomorphic) であるといわれる．いいかえれば接続行列または隣接行列において，適当に行と列を入れかえることによって二つの行列が同じものになることである．与えられた二つのグラフが同形であるか否かを具体的に判定するための能率の良い算法は，特殊な場合[15]を除いて今のところ知られていない．

グラフ $\mathbf{G}=(V,E)$ の枝 $e\in E$ を**取り除く**というのは，$\mathbf{G}$ から $\mathbf{G}'=(V,E')$; $E'=E-\{e\}$ なる新しいグラフ $\mathbf{G}'$ を得る操作のことである．逆に枝 $e\notin E$ を**付加する**というのは，$\mathbf{G}$ から $\mathbf{G}'=(V,E')$; $E'=E\cup\{e\}$ なる新しいグラフ $\mathbf{G}'$ を得る操作のことである．与えられたグラフからいくつかの枝を取り除くこ

* ここで“描くことができない”というのは，“行と列のうちの片方が枝に他方が節点に対応したグラフが存在しない”という意味であって，行と列の両方を節点に対応させれば 2 部グラフとして表現できる (2.7 節参照) ことに注意されたい．

とによって得られるグラフを G の**部分グラフ**(subgraph), 逆にいくつかの枝を付加することによって得られるグラフを G の**スーパーグラフ** (supergraph) という. グラフ $G=(V, E)$ の節点の部分集合 $A \subset V$ に対して, 両端点が共に A に属している枝だけを残して, 他の枝をすべて取り除いて (孤立節点は考えない) 得られるグラフを $G(A)$で表わし, これを**セクショングラフ** (section graph) という. $G(A)$ が完全グラフとなるとき, A を**クリーク** (clique) という.

枝と同様に, 節点を取り除いたり, 付加したりすることもできる. 最も簡単な方法は, 次数2の節点に接続している二つの枝をまとめて一つの枝とみなした後にその節点を取り除く操作である. 逆にある枝の上に一つの節点を挿入することによって二つの枝が直列に並んだものと考える操作もある. このような意味において, いくつかの節点を取り除く, あるいは付加することによって同形となり得る二つのグラフは, 互いに**同相** (homeomorphic) であるといわれる. 例えば図-5の二つのグラフは互いに同相である.

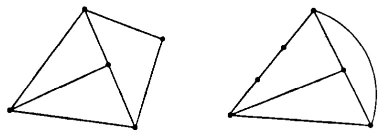

図-5 互いに同相なグラフ

2.3 連結性

グラフの $l+1$ 個の節点 $\{v_0, v_1, \cdots, v_l\}$ と l 個の枝 $\{e_1, e_2, \cdots, e_l\}$ を交互に並べた系列

$$P=(v_0, e_1, v_1, e_2, \cdots, e_l, v_l) \tag{2・6}$$

において, $e_i=(v_{i-1}, v_i)$; $i=1, 2, \cdots, l$ であるとき, P は長さ l の**道** (path) と呼ばれる. 特に $v_0=v_l$ のときには, P は長さ l の**閉路** (circuit) と呼ばれる. 同じ枝を2度以上通らないような道は**単純** (simple) な道, 同じ節点を2度以上通らないような道は**初等的** (elementary) な道といわれる. 閉路についても同様である. 特に証明するまでもなく, 初等的な道 (閉路) は単純な道 (閉路) でもある.

上に定義した道の概念を用いて, グラフの連結性を議論することができる. 如何なる節点の対に対してもその二点間を結ぶ道が存在するようなグラフを**連結グラフ** (connected graph) という. グラフ G においてその連結な極大セクショングラフの各々を G の**連結成分** (connected component) という. 連結グラフ G の枝の集合 E を E_1 と E_2 ($E_1 \cap E_2=\phi$, $E_1 \cup E_2=E$) に分割して, E_1 の端点の集合を V_1, E_2 の端点の集合を V_2 とする. いま $V_1 \cap V_2$ がただ一点から成るような分割 (E_1, E_2) が存在するとき, G は**可分** (separable) であるという. 非連結グラフは可分であるとみなす. 一般のグラフ G において, 極大な非可分部分グラフを G の**非可分成分** (nonseparable component) という. 例えば図-6のグラフは三つの非可分成分に分解される. 一般にグラフの連結成分, 非可分成分は一意に定まり, 非可分成分への分割は連結成分への分割をさらに細かく分けたものとなっている.

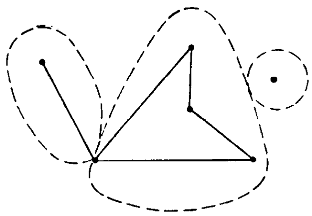

図-6 グラフの非可分成分

グラフのすべての枝を通るような単純な道あるいは閉路のことを, それぞれ**オイラー路** (Euler path) あるいは**オイラー閉路** (Euler circuit) という. オイラー (閉) 路に沿って筆を運んでいくと, 途中でとぎれることなくグラフが描けるところから, オイラー(閉)路を有するグラフは**一筆書き可能** (unicursal) であるともいわれる. グラフの一筆書き可能性は, 下記の有名な定理によって, 単に節点の次数を調べるだけで容易に判定できる.

(i) グラフ G にオイラー閉路が存在するための必要十分条件は, G が連結でかつすべての節点の次数が偶数であることである.

(ii) グラフ G にオイラー路が存在するための必要十分条件は, G が連結でかつ0個あるいは2個の節点の次数が奇数でその他のすべての節点の次数が偶数であることである.

グラフのすべての節点を含む初等的な道あるいは閉路のことを**ハミルトン路** (Hamilton path) あるいは**ハミルトン閉路** (Hamilton circuit) という. ハミルトン (閉) 路については, オイラー (閉) 路の場合と異なって, その存在条件や作図法などについて実際に役立つことはあまり知られていない.

ハミルトン閉路の概念と密接な関連をもつものとして**巡回セールスマン問題**がある[16]. これはある都市から出発して, 指定された都市の各々をちょうど一回だけ訪問しながら元の都市に戻るという条件のもとで,

最短の径路を求めよ，という問題である．都市を節点に，都市と都市とを結ぶ道路を枝に対応させたグラフを考え，さらに道路の長さをこれに対応した枝の重みとすれば，巡回セールスマン問題は重みの和最小のハミルトン閉路を求める問題に他ならない．この問題の解法としては，すべてのハミルトン閉路を“虱潰し”的にあたる方法に毛がはえた程度のものしか知られていない．従って大きなグラフの巡回セールスマン問題は計算機を利用しても解くことは不可能に近い（例えば節点数 20 の完全グラフには $(20-1)!/2 \simeq 6\times10^{16}$ 個のハミルトン閉路が存在する．

グラフ $\mathbf{G}=(V,E)$ の節点の集合 V を二つの部分 V_1, V_2 に分割（$V_1\cap V_2=\phi$, $V_1\cup V_2=V$）したとき，一方の端点は V_1 に他方は V_2 に属するような枝の集合をグラフ $\mathbf{G}$ の**カットセット** (cutset) という．一つの節点に接続している枝の集合は，カットセットの特殊な場合である．あるカットセットにおいて，真部分集合としてカットセットを含まないとき，これを**初等的** (elementary) なカットセットという．カットセットは先に定義した閉路と双対な（**2.5** 節参照）関係にある重要な概念である．**図-7** のグラフにおいて $P=(v_1, e_2, v_2, e_3, v_3, e_4, v_4, e_1, v_1)$ と $C=\{e_2, e_4, e_5\}$ はそれぞれ閉路とカットセットの例である．

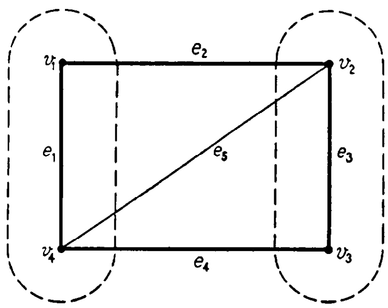

図-7 閉路とカットセット

2.4 木と補木

連結グラフ $\mathbf{G}=(V,E)$ の部分グラフ $\mathbf{G}'=(V,F)$; $F\subset E$ において，$\mathbf{G}'$ が連結グラフでかつ閉路を含まないとき，F を $\mathbf{G}$ の**木** (tree) という．$\mathbf{G}$ が非連結のときは個々の連結成分の木の集まりを木という（これを**森** (forest) と呼んで区別することもある）．また，$\mathbf{G}'$ が $\mathbf{G}$ のカットセットを含まず，かつ任意の枝を付加するとこの性質を満たさないとき，F を $\mathbf{G}$ の**補木** (cotree) という．木と補木の関係を表わすものとして，“木の補集合は補木で，補木の補集合は木である”ことが良く知られている．

グラフの節点数，枝の数，連結成分の数をそれぞれ n, m, k としたとき，

$$\begin{cases}\rho=n-k\\ \mu=m-n+k\end{cases} \tag{2.7}$$

で与えられる整数 ρ, μ をそれぞれグラフの**階数** (rank), **零度** (nullity) という．ρ, μ と上に述べた木，補木の概念は“木，補木の枝の数はそれぞれ ρ, μ に等しい”という定理によって密接に結びついている．

グラフの一対の木，補木を固定したとき，一つの木の枝といくつかの補木の枝とで一つのカットセットが形成され，しかもこれは一意に定まる．同様にして，各々の補木の枝は補木の枝を一つだけ含む閉路を一意に定める．このようにして定められる ρ 個のカットセット，μ 個の閉路をそれぞれ**基本カットセット**，**基本閉路**という．**図-8** は木（太線）$T=\{a, c, d, f\}$ と補木（細線）$\overline{T}=\{b, e, g\}$ の対の例を示している．このとき T に対応する基本カットセットは

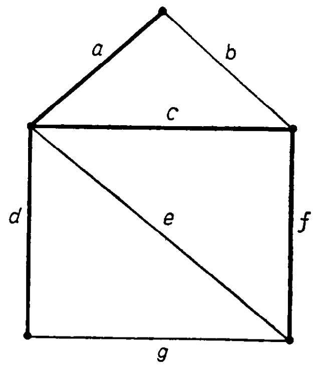

図-8 木と補木

$\{a, b\}$, $\{c, b, e, g\}$, $\{d, g\}$, $\{f, e, g\}$

である．また $\overline{T}$ に対応する基本閉路は

$\{b, a, c\}$, $\{e, c, f\}$, $\{g, c, d, f\}$

である．

グラフの木を求めるには，閉路を形成しないように枝を一つずつ付加していけば良い．適切な算法上の工夫をすることによって，m に比例する手間で一つの木が求められることが知られている[17]．グラフの木を一つ求めれば，結果的に個々の連結成分を求めたことになるので，グラフの連結成分への分割も m に比例する手間でできることになる[18]．非可分成分への分割も，木を求める操作を工夫——**深さ優先探索** (depth first search) という方法を用いる——すれば，やはり m に

比例する手間で十分である[18]．枝の重みが与えられたグラフにおいて，重みの和最小の木——これを**最小木** (minimum tree) という——を求める問題は，実際への応用面で重要である．グラフの最小木（あるいは最大木）を求める場合，n^2 に比例する手間の算法[19]が最善である．

最小木を求める問題と似ていながら，はるかに困難であるという意味で，**スタイナー木** (Steiner tree) を求める問題は有名である[20]．スタイナー木問題とは，グラフ $\mathbf{G}=(V,E)$ と節点の部分集合 $V'\subset V$ に対して，V' に含まれるどの二点の間も道で結ぶことができるような部分グラフのうち，枝の重み最小のものを求めよ，という問題である．$V-V'$ の節点が V' の節点に連結していてもいなくても良いというところが，問題を難しくている原因である．

2.5 双対グラフと平面グラフ

二つのグラフ $\mathbf{G}_1$, $\mathbf{G}_2$ の枝の集合の間に一対一対応が存在し，その対応のもとで $\mathbf{G}_1$ の閉路と $\mathbf{G}_2$ のカットセットとが対応するとき，$\mathbf{G}_1$ と $\mathbf{G}_2$ は互いに**双対なグラフ** (dual graph) であるといわれる．この概念は「閉路とカットセットとは双対な概念である」などというときの**双対性** (duality) の概念と混同され易いようである．"双対なグラフ" というのはグラフ理論の中の概念であり，"双対性" というのはグラフ理論を外から眺めたときの話である[21]，ということに注意されたい

平面上に枝を交差させずに描くことのできるグラフを**平面グラフ** (planar graph) という．グラフの平面描写の問題はプリント基板や集積回路の配線と関連して重要である．グラフが平面グラフであるか否かを判定し，そうである場合には具体的な平面描写を求める問題については，最近能率の良い算法が確立された[22]が，平面でないグラフから最小数の枝を取り除いて平面グラフを作る，という問題については殆んど何も判っていない．

グラフが平面であるための必要十分条件はいろいろ知られているが，代表的なものとして下記がある．

（i） 平面上に枝の交差なしに描ける．

（ii） 双対グラフが存在する．

（iii） 部分グラフとして，図-3 のいずれかのグラフ——これらを**クラトフスキグラフ** (Kuratowski graph) という——と同相なものを含まない．

上の諸条件は互いに等価であるが，その証明は厄介であるので，例えば文献 23) を参照されたい．ただし

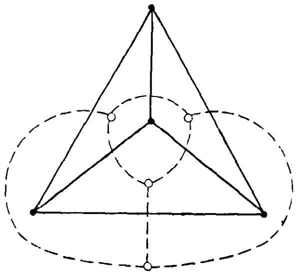

図-9　双対なグラフ

条件（i）から（ii）の導入は直観的に理解できる．すなわちグラフの平面描写が与えられれば，**図-9** の点線で示すような作図によって，これと双対なグラフが得られる．

2.6 彩　色　数

グラフ $\mathbf{G}$ の節点の集合を γ 個の部分に分割し，同一の部分の中の節点が隣接していないようにできる γ の最小値を $\mathbf{G}$ の**節点彩色数**あるいは単に**彩色数** (chromatic number) と称し，$\gamma(\mathbf{G})$ と記す．これは隣接節点が異なる色で塗られるように節点を色分けする際に必要な色の最小数と解釈される．

グラフ $\mathbf{G}$ の含むクリークの内，最大なクリークの大きさ（節点数）を $\mathbf{G}$ の**クリーク数**(clique number) と称し，$\omega(\mathbf{G})$ と記す．定義から，$\gamma(\mathbf{G})\geqq\omega(\mathbf{G})$ が成立することは明らかであろう．特に $\gamma(\mathbf{G})=\omega(\mathbf{G})$ が成立するグラフは **γ-完全** (γ-perfect) であるといわれる．

グラフ $\mathbf{G}$ の長さ 4 以上の閉路において連続していない二つの節点の対を結ぶ枝をこの閉路の**弦** (chord) という．任意の長さ 4 以上の閉路が少なくとも一つの弦を含むようなグラフを**三角化グラフ** (triangulated graph)という．これに関しては，次の定理が証明されている[24]．

"三角化グラフは γ-完全である"

三角化グラフはスパース行列手法への応用に関連して，最近深く研究され[25][26]，クリーク数（＝彩色数）は比較的簡単な手続きによって求められることが知られている．

グラフ $\mathbf{G}$ の節点の集合 $\{v_1, v_2, \cdots, v_n\}$ と実直線上の区間の集合 $\{I_1, I_2, \cdots, I_n\}$ の間に一対一対応が存在して，任意の 2 節点 v_i, v_j を端点とする枝が存在するとき，しかもそのときに限り，対応する区間の対 I_i, I_j が重なるようにできるならば，$\mathbf{G}$ は**区間グラフ** (interval graph) であるといわれる．これに関しては，

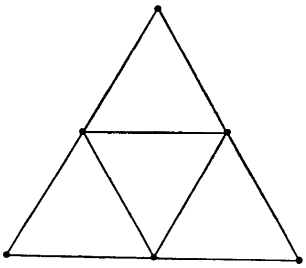

図-10　三角化グラフ

“区間グラフは三角化グラフである”
という事実が知られている[24]．ただし，この逆は必ずしも成立しないことに注意されたい．例えば図-10 のグラフは三角化グラフであるが，区間グラフではない．図-11 は区間の集合（a）とこれに対応している区間グラフ（b）の例を示している．区間グラフの概念はプリント基板や集積回路の配線問題への応用に関連して重要である．

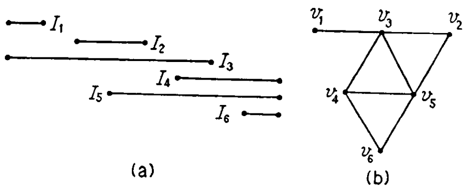

図-11　Set of intervals (a) and its graph (b)

2.7　被覆とマッチング

グラフ G の枝の部分集合 F において，各々の節点が F のどれかの枝の端点となるとき，F は G の**被覆**(covering) という．特に F の枝の数が最小のものを**最小被覆**という．図-12 のグラフにおける太線の枝の集合はその一例である．グラフの最小被覆は n^3（n は節点の数）に比例する手間で求められることが知られている[27]．

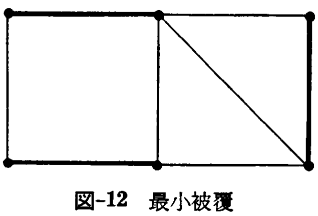

図-12　最小被覆

被覆の概念は接続行列の上で解釈することもできる．Fを列の部分集合とし，各々の行が少なくとも一つの F の列の中に“1”要素をもつものが，上に述べた被覆に相当する．このような意味において，被覆の概念は一般に“0”要素と“1”要素だけから成る行列に対しても自然な形で拡張できる．行列の被覆は，行が節点を列が広義の枝を表わすとすれば，ハイパーグラフの被覆であるといっても良い．しかし一般の行列の最小被覆を求める問題は，グラフの場合ほど簡単でなく虱潰しに近い算法しか知られていない．

グラフの節点に着目した被覆の概念もある．すなわち接点の部分集合 W において，各々の枝が少なくとも一つの W の節点に接続しているとき，W を**節点被覆**と呼ぶ．この場合，前に述べた被覆を**枝被覆**と呼んで区別するのが普通である．

グラフ G の枝の部分集合 F において，F のどの二つの枝も共通端点をもたないとき，F を G の**マッチング** (matching) という．マッチングの理論は特に2部グラフを対象として，かなり体系的に研究されている分野である．以下グラフ $\mathbf{G}=(V,E)$ は2部グラフとし，V の分割を (V_1, V_2) とする．G のマッチング $F\subset E$ の中で，F の枝の数が最大のものを**最大マッチング**，すべての節点が F のどれかの枝の端点であるものを**完全マッチング**という．明らかに，Fがマッチングなら，$|F|\leqq\min\{|V_1|,|V_2|\}$ であり，また G の上に完全マッチングが存在するならそれは最大マッチングであり，かつそのとき $|V_1|=|V_2|$ でなければならない．図-13 に完全マッチング（太線）の例を示す．2部グラフのマッチングと節点被覆の間には，“最大マッチングの要素数（枝の数）は最小節点被覆の要素数（節点の数）に等しい”という重要な関係 (König-Egerváry の定理) がある．2部グラフの最大マッチングは $n^{5/2}$ に比例する手間で求められることが知られている[28]．

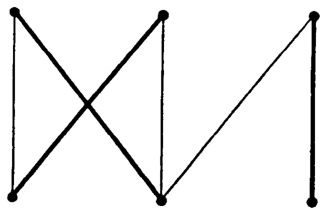

図-13　完全マッチング

2部グラフは行列の（各要素が0か否かという意味での）構造を表わしていると考えることもできる．この場合，節点の分割 (V_1, V_2) の片方が行に，他方が列に対応している．$|V_1|=|V_2|$ となる2部グラフにおいて完全マッチングが存在すれば，しかもそのときに限り，対応する正方行列の行と列の適当な置換によ

って対角項に非0要素を並べることができる．この事実は先に述べた König-Egerváry の定理の一つの解釈である．

2.8 有向グラフ

これまで扱ったグラフにおいては，枝の方向すなわち両端点の順序は問題にしなかった．しかし，目的によっては枝の方向を考慮した方が便利なこともある．枝が方向性をもつグラフを**有向グラフ** (directed graph あるいは digraph) といい，これに対して方向性をもたないグラフを**無向グラフ** (undirected graph) と呼んで区別することもある．例えば**図-14**(a) の論理回路図は，信号の流れを考慮して同図 (b) の有向グラフで表わされる．

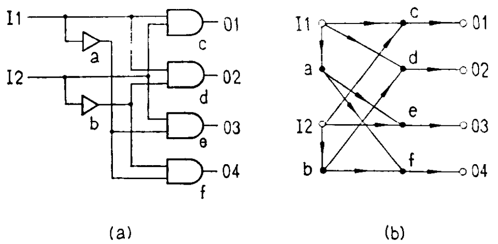

図-14 論理回路図(a) とその有向グラフ表現(b)

有向グラフにおいて，枝 $e=(u,v)$ が端点 u から端点 v に向うように方向付けされているとき，u, v をそれぞれ e の**始点**，**終点**と呼んで区別する．節点 v がこれに接続しているすべての枝の始点あるいは終点であるとき，v をそれぞれ**入力節点** (source node), **出力節点** (sink node) と呼ぶ．有向グラフの特殊な場合として，入力節点と出力節点を一つずつもつグラフ——これを **SEC グラフ** (single-entry single-exit connected graph) という——は実際面での応用範囲が広い．

有向グラフを隣接行列で表現するには，式 (2・1) において枝 $e=(i,j)$ の始点が i のときだけ $a_{ij}=1$ となるように定義を変更すれば良い．また接続行列で表現するには，式 (2・3) において節点 i が枝 j の始点(終点) であるとき $d_{ij}=1$ ($d_{ij}=-1$) と変更すれば良い．

2.3 節で定義した道，閉路，カットセットなどの概念についても，方向性を問題とすることが多い．式 (2・6)の系列 P において，各々の枝 e_i ; $i=1\sim l$ の始点が v_{i-1}, 終点が v_i となるとき，P を**有向道**といい，v_0, v_l をそれぞれ P の始点，終点という．特に，$v_0=v_l$ のときは，P を**有向閉路** (directed circuit あるいは cycle) と呼ぶ．また，グラフ $\mathbf{G}=(V,E)$ の節点の集合 V の分割 (V_1, V_2) に対応するカットセット $F\subset E$ において，F のすべての枝の始点(終点) が V_1 に終点 (始点) が V_2 に含まれるとき，F を $\mathbf{G}$ の**有向カットセット**と呼ぶ．

無向グラフにおいては，連結成分を非可分成分に分解することが問題となったが，有向グラフでは別の角度からの分解が重要である．有向グラフ $\mathbf{G}=(V,E)$ において，どの節点の順序対 (u,v) に対しても，u から v に致る有向道が存在するとき，$\mathbf{G}$ は**強連結**(strongly connected) であるという．また $\mathbf{G}$ のセクショングラフ $\mathbf{G}(A)$; $A\subset V$ において，$\mathbf{G}(A)$ が強連結かつ，任意の A 以外の節点 $v\in V-A$ に対して $\mathbf{G}(A\cup\{v\})$ が強連結でないとき，$\mathbf{G}(A)$ を $\mathbf{G}$ の**強連結成分**という．有向グラフを強連結成分に分解したとき，節点の集合もこれに対応して分割されるわけであるが，異なった二つの成分に属する節点対を結ぶ枝が存在することに注意されたい．これらの枝が強連結成分相互の半順序関係を定める(**図-15** 参照)．有向グラフを強連結成分に分解する問題も，深さ優先探索を利用して m (枝の数) に比例する手間で解ける[18]．

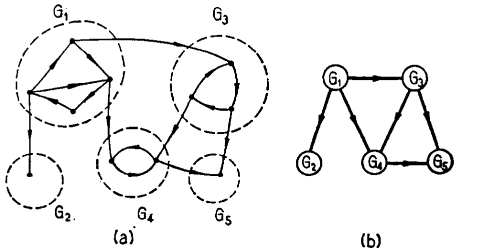

図-15 有向グラフの強連結成分(a) とそれらの半順序関係(b)

有向グラフは2部グラフと対応関係をもっている．有向グラフ $\mathbf{G}=(V,E)$ の節点の集合 $V=\{v_1, v_2, \cdots, v_n\}$ から，これと一対一対応をもつもう一つの節点の集合 $V'=\{v_1', v_2', \cdots, v_n'\}$ を考え，つぎのような2部グラフ $\mathbf{G}'=(V\cup V', E')$ を作ることができる．すなわち $\mathbf{G}$ の枝$(v_i, v_j)\in E$ の始点が v_i であるとき，しかもそのときに限り，$\mathbf{G}'$ は枝(v_i, v_j') をもつように $\mathbf{G}'$ の接続関係を定義する (**図-16**(次頁参照))．これによって $\mathbf{G}$ から $\mathbf{G}'$ は一意に定まり，逆に V の節点と V' の節点との対応関係を与えれば，$\mathbf{G}'$ から $\mathbf{G}$ も一意に定まる．有向グラフを強連結成分へ分解する問題の性質や具体的算法についての2部グラフの立場からの研究もある[29),30)]．

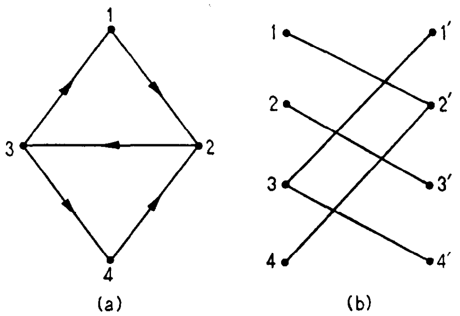

図-16 有向グラフ(a)とこれに対応した2部グラフ(b)

有向グラフに関連して，**最小帰還枝（節点）集合** (minimum feedback arc (vertex) set) を求める問題がある．これは枝（節点）を取り除いて，与えられた有向グラフの有向閉路をすべて消滅させるために必要な最小数の枝（節点）の組を求める問題である[31]．ここで“節点を取り除く”というのは，“その節点に接続しているすべての枝を取り除く”ことである．この問題は，与えられた系に対してなるべく元数の少ない連立方程式をたてるという問題に相当し，興味深いものであるが能率の良い算法は知られていない．

最後に，種々のネットワークの解析，設計において基本的役割を果たす**最短路** (shortest path) 問題にふれておこう．これは枝に重みづけされたグラフにおいて，指定された2節点間を結ぶ道（有向グラフの場合は有向道）の中で，重みの和が最小のものを求める問題である．この問題は能率の良い算法を求める立場から幅広く研究されており，n^2 に比例する手数で最短路が求められる Dijkstra 法[32]が最善とされている．また，すべての節点対の間の最短路を求める問題については，Floyd 法[33]——基本演算の順序は異なるが総演算回数はこれと同じになる算法は種々考えられる——が最善とされている．類似な問題として，重みの和最大の（初等的な）有向路を求める問題があり，これは日程計画問題において重要な役割を果たす．有向閉路を含まない有向グラフにおいては，**最長路** (longest path) 問題も最短路問題の場合と同等の手法によって解ける．しかし無向グラフや，有向グラフでも有向閉路を含む場合は，すべての道を虱潰しに当る方法以外に，最長路を求める算法は知られていない．この場合，最短路問題と異なり，最長路問題は凸な計画問題にならないことに注意されたい．

参考文献

1) O. Ore: Theory of Graphs, Colloquium Publications, **38**, American Mathematical Society, Providence (1962).
2) C. Berge: The Theory of Graphs and its Applications (trans. by A. Doig), Methuen, London (1962).
3) F. Harary: Graph Theory, Addison-Wesley, Reading (1969).
4) C. Berge: Graphes et Hypergraphes, Dunod, Paris (1971).
5) 尾崎弘，白川功：グラフとネットワークの理論，コロナ社，(1973).
6) N. Deo: Graph Theory with Applications to Engineering and Computer Science, Prentice-Hall, Englewood Cliffs (1974).
7) R.G. Busacker & T.L. Saaty: Finite Graphs and Networks—An Introduction with Applications, McGraw-Hill, New York (1965).
8) S. Seshu & M.B. Reed: Linear Graph and Electrical Networks, Addison-Wesley, Reading (1961).
9) W.H. Kim & R.T. Chen: Topological Analysis and Synthesis of Communication Networks, Columbia Univ. Press, New York (1962).
10) L.R. Ford, Jr. & D.R. Fulkerson: Flows in Networks, Princeton Univ. Press, Princeton (1962).
11) M. Iri: Network Flow, Transportation and Scheduling Theory and Algorithms, Academic Press, New York (1969).
12) T.C. Hu: Integer Programming and Network Flows, Addison-Wesley, Reading (1969).
13) H. Frnk & I. T. Frisch: Communication, Transmission, and Transportation Networks, Addison-Wesley, Reading (1971).
14) E. L. Lawler: Cutsets and Partitions of Hypergraphs, Networks, Vol. 3, pp. 275～285 (1973).
15) J. Hopcroft & R. Tarjan: A V^2 Algorithm for Determining Isomorphism of Planer Graphs, Information Processing Letters, Vol. 1, pp. 32～34 (1971).
16) M. Bellmore & G. L. Nemhauser: The Traveling Salesman Problem: A Survey, Operations Res., Vol. 16, pp. 538～558 (1968).
17) J. J. Seppänen: Algorithm 399: Spanning Tree, Comm. ACM, Vol. 13, No. 10, pp. 621～622 (1970).
18) R. Tarjan: Depth-First Search and Linear Graph Algorithms, SIAM J. Comput., Vol. 1, No. 2, pp. 146～160 (1972).

19) V. K. M. Whitney : Algorithm 422 : Minimal Spanning Tree, Comm. ACM, Vol. 15, No. 4, p. 273 (1972).
20) S. K. Chang : The Generation of Minimal Trees with a Steiner Topology, J. ACM, Vol. 19, No. 4, pp. 699〜711 (1972).
21) M. Iri : Metatheoretical Considerations on Duality, RAAG Research Notes, 3 rd Series, No. 124 (1968).
22) J. Hopcroft & R. Tarjan : Efficient Planarity Testing, J. ACM, Vol. 21, No. 4, pp. 549〜568 (1974).
23) W. T. Tutte : Matroids and Graphs, Trans. Am. Math. Soc., Vol. 90, pp. 527〜552 (1959).
24) C. Berge : Some Classes of Perfect Graphs, in "Graph Theory and Theoretical Physics (F. Harary, Ed.)", pp. 155〜166, Academic Press, New York (1967).
25) D. J. Rose : Triangulated Graphs and the Elimination Process, J. Math. Anal. and Appl., Vol. 32, pp. 597〜609 (1970).
26) T. Ohtsuki, L. Cheung & T. Fujisawa : On Minimal Triangulation of a Graph, Memorandum No. ERL-M351, Univ. of California, Berkeley (1972).
27) J. Edmonds : Paths, Trees, and Flowers, Canadian J. of Math., Vol. 17, pp. 449〜467 (1965).
28) J. E. Hopcroft and R. M. Karp : A $n^{5/2}$ Algorithm for Maximum Matchings in Bipartite Graphs, IEEE Conf. Record of the 12 th. Annual Symp. on Switching and Automata Theory, pp. 122〜125 (1971).
29) A. L. Dulmage & N. S. Mendelsohn : A Structure Theory of Bipartite Graphs of Finite Exterior Dimension, Trans. Roy. Soc. Can., Sec. III, Vol. 53, pp. 1〜13 (1959).
30) A. L. Dulmage & N. S. Mendelsohn : Two Algorithms for Bipartite Graphs, SIAM J., Vol. 11, No. 1, pp. 183〜194 (1963).
31) A. Lempel : Minimum Feedback Arc and Vertex Sets of a Directed Graph, IEEE Trans. Circuit Theory, Vol. CT-13, No. 4, 399〜403 (1966).
32) E. W. Dijkstra : A Note on Two Problems in Connection with Graphs, Numerische Math., Vol. 1, pp. 269〜271 (1959).
33) R. W. Floyd : Algorithm 97 : Shortest Path, Comm. ACM, Vol. 5, p. 345 (1962).

(昭和 50 年 2 月 3 日受付)
(昭和 50 年 2 月 22 日再受付)